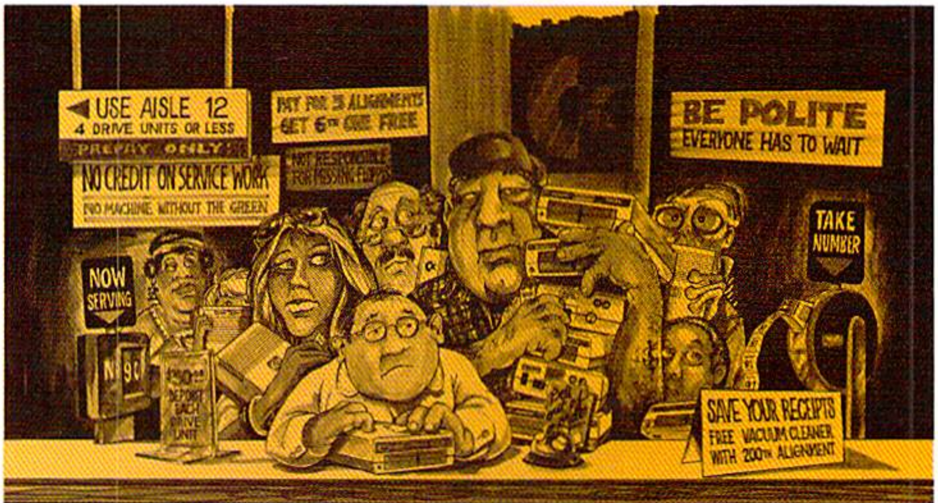
COMMODORE USERS

Ahoy!

48355

$2.75/ CAN. $3.25 FEB. 1986





Who needs this?

When you can solve disk drive alignment problems in 60 minutes with the CSM program.

Disk drive alignment problems?
Drive out of alignment again?
Tired of waiting two weeks or more to get your drive fixed??

WE HAVE THE ANSWER ! !

With the 1541 DISK DRIVE ALIGNMENT PROGRAM you can align the drive yourself in an hour or so. Not only that, you can do it at home AND no special equipment is required. Anyone with average mechanical skills can do it!!

Read What Compute!'s Gazette had to say. (Oct., 1984)

". . . with 1541 Disk Drive Alignment *from CSM Software, you can fix it* [the disk drive] *yourself in an hour or so and the program will pay for itself the first time you use it...No technical expertise is required to accomplish the alignment procedures, and the manual accompanying the program thoroughly describes the procedures."*

1541 DISK DRIVE ALIGNMENT PROGRAM - VERSION 2.0 - *$44.95 plus shipping*

CARTRIDGE BACKER V2.0

IMPROVED AND UPDATED!!

The best cartridge - to disk backup system is now even better. The software has been enhanced to back up many more cartridges and disk programs. Cartridges will run from the disk as-is and do not require any ROM or RAM expansion boards. VERSION 2.0 software update is available to ALL prior purchasers of CARTRIDGE BACKER and CARTRIDGE CRACKER (from BEAR).

Complete system
(includes software and expansion board) —

$54.95 plus shipping

Software update only (call for details) —

$14.00 plus shipping

PROGRAM PROTECTION MANUAL FOR THE C-64 VOLUME II

Not just a third edition — a new and up-to-date manual covering the latest advances in program protection. Over 275 pages of valuable information. Topics include:

- *ENCRYPTION AND DECRYPTION*
- *UNDOCUMENTED OPCODES*
- *GCR RECORDING*
- *CUSTOM DOS ROUTINES*
- *CARTRIDGES AND EPROMS*
- *PROTECTING YOUR OWN SOFTWARE*
- *TIPS FROM EXPERTS ON PROTECTION, BACKUP & MORE*
- *DISK INCLUDED*

$34.95 plus shipping

★ ★ NEW PRODUCT ★ ★

EPROM PROGRAMMERS HANDBOOK

Finally, a reference book written for the average user. Until now EPROM programming was simply too difficult for most people. The EPROM PROGRAMMERS HANDBOOK will take the mystery out of putting your favorite programs on cartridge. Learn to create your own cartridges and customize your computer and disk drive. Discover the power of programming your own EPROMs.

- *Covers beginning thru advanced topics.*
- *Simplifies EPROM programming so that anyone can do it.*
- *Perfect companion for the PROMENADE and other EPROM programmers.*
- *Create your own auto-start cartridges, custom KERNAL, modified DOS and specialized EPROMs.*
- *Put many programs on a single cartridge with MENU-MAKER.*
- *Disk includes ready-to-use routines and helpful utilities.*

$32.95 plus shipping

PROGRAM PROTECTION MANUAL FOR THE C-64 VOLUME I

- *A MUST FOR BEGINNERS*
- *THE PERFECT INTRODUCTION TO PROGRAM PROTECTION*
- *DISK INCLUDED*

$29.95 plus shipping

VISA AND MASTERCARD ACCEPTED
Shipping $3.50 per item in U.S.; foreign orders extra

SNAPSHOT 64

SNAPSHOT 64 is a unique and exciting utility that virtually takes a picture of your computer's memory. SNAPSHOT 64 then saves that 'snapshot' to disk and automatically creates an auto-boot loader for the program. Once saved to disk the program may be restarted at the exact same point at which it was interrupted!!!

SNAPSHOT 64 is the perfect answer for those looking for the ultimate backup utility. Think of it, being able to stop most any program after the protection check and then being able to resume the program at the same point, totally bypassing the protection check.

- **MOST EFFECTIVE UTILITY OF ITS KIND** - *EVEN MORE EFFECTIVE THAN THOSE UTILITIES THAT CLAIM TO "COPY ALL MEMORY-RESIDENT SOFTWARE"*
- *INVISIBLE TO SOFTWARE — CANNOT BE DETECTED*
- *SNAPSHOT FILES RUN BY THEMSELVES — THE BACKUP COPY WILL RUN AS-IS, WITHOUT THE SNAPSHOT CARTRIDGE*
- *WORKS MUCH FASTER THAN OTHER SIMILAR UTILITIES*
- *EASY TO USE — NO COMPLICATED INSTRUCTIONS — EVEN A CHILD CAN DO IT*
- *SNAPSHOT FILES ARE COMPATIBLE WITH FAST LOAD UTILITIES*
- *SNAPSHOT FILES WILL LOAD FROM ALL COMMODORE COMPATIBLE DISK DRIVES*
- *STORE MANY DIFFERENT SNAPSHOTed FILES ON A SINGLE DISK*
- *FULLY SELF-CONTAINED ON CARTRIDGE — NO DISK TO HASSLE WITH*
- **EXCLUSIVE FEATURE:** *THE CODE INSPECTOR — ALLOWS MOST PROGRAMS TO BE STOPPED, EXAMINED, MODIFIED AND RESUMED WHILE STILL IN MEMORY!!*
- *NO PARAMETERS NEEDED — AS OFTEN REQUIRED BY OTHER UTILITIES*
- *COMPLETE AND THOROUGH MANUAL IS INCLUDED*
- *SOLD FOR ARCHIVAL PURPOSES ONLY*
- *RAVE REVIEWS IN:* INFO 64, MIDNIGHT GAZETTE & PROGRAM PROTECTION NEWSLETTER

SNAPSHOT 64™
$49.95 + $3.50 SHIPPING

C= *is a registered trademark of Commodore Business Machines, Inc.*

Available through your local software dealer or call:

CSM SOFTWARE, INC.

POST OFFICE BOX 563, CROWN POINT IN. 46307, PHONE (219) 663-4335

Reader Service No. 165

Ahoy!

CONTENTS

DEPARTMENTS

FEATURES

*Includes programs: *Extended Background* and *Windows* for the C-64

**Includes program: *ELMAC Plays NIM* for the C-64

***Includes program: *Turnkey 64* for the C-64

****Includes program: *MOVECHRS* for the C-64

*****Includes program: *Compuphobia* for the C-64

******Includes program: *1541 Alignment Check* for the C-64

PROGRAMS

Cover art by James Regan; art inside screens by Tom Cushwa, Javier Romero

ISSUE NO. 26 FEBRUARY 1986

Ahoy! (#8750-4383) is published monthly by Ion International Inc., 45 W. 34th St., Suite 407, New York, NY, 10001. Subscription rate: 12 issues for $21.95, 24 issues for $41.75 (Canada $29.25 and $54.25 respectively). Second class postage paid at New York, NY 10001 and additional mailing offices.

Postmaster, send address changes to *Ahoy!*, 45 W. 34th Street, Suite 407, New York, NY 10001. Direct all address changes or matters concerning your subscription to *Ahoy!*, P.O. Box #341, Mt. Morris, IL 61054. All editorial inquiries and software and hardware to be reviewed should be sent to *Ahoy!*, 45 W. 34th St., Suite 407, New York, NY 10001.

DISTRIBUTOR PRICES ARE NOW ACCESSIBLE!

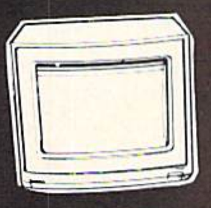
HOLIDAY SAVINGS



Gemini SG10 PRINTER

$215

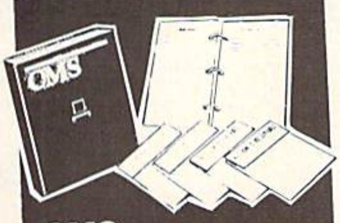
•Including: Shipping, Handling and Insurance — No Extra Charges



Goldstar 13" COLOR COMPOSITE MONITOR

•Cables Included •1 Yr. Mfr. Warranty

$124.95



CMS
FOR 128 Commodore PERSONAL COMPUTER SPREAD SHEET BOOK PACKAGE

$139.95

PRINTER SPECIALS FROM CCA

GEMINI STAR 10X

$149 Limited Supply

STAR POWER TYPE

•Letter Quality •18 CPS •13" Carriage with Comm. Interface

$289.95

MODEMS



Mighty Mo $64.95
Commodore 1660
Call for Price
Commodore 1670
Call for Price

JOYSTICKS



GIM Professional Joystick

•2 Fire Buttons •Contoured Handle •Lifetime Guarantee •Comparable to 3-Way

$9.95

WICO JOYSTICKS

The Boss	The Bat	3-Way
$12.88	$14.88	$19.88

AND MORE!

COMPUTER CENTERS OF AMERICA!!!!

ACTIVISION



Pet Person (Little People) $24.88
Fast Trax $24.88
Hacker $14.88
Alcazar $14.88
Countdown to Shutdown $14.88

ACCESS



Neutral Zone D-T $20.95
Spritemaster D-T $20.95
Beachhead D-T $20.95
Master Composer D $23.95

PROFESSIONAL

Fleet System 2
The Largest Spell Checker Available For Your Computer
90,000 Words on C-64/128
70,000 Words on Atari

$59.95



penguin software™

Quest $19.99
Transylvania $19.99
Sword of Kadash $19.99
Xyphus $19.99
Graphics Magician Picture Painter $21.99

WE'RE PROUD OF OUR PRICES — SATISFACTION GUARANTEED

Check, Money Order, MC or VISA accepted / No additional charge for MasterCard and Visa / Call for shipping and handling information / NYS residents add applicable sales tax / Prices and availability are subject to change without notice / All factory fresh merchandise with manufacturer's warranty. Dealers Welcome. Call for recent price reductions and new rebate information. Plus shipping and handling. No returns without return authorization number.

COMPUTER Centers of America Mail Order LTD.

VIEW FROM THE BRIDGE

By now the ad on the facing page has snagged your eye, and you know the story behind the banner headline on this month's cover. We invite our readers to share our joy in being rated number one by *INFO* Magazine, the highly-respected *Consumer Reports* of the Commodore marketplace.

But gloating didn't get us where we are—and it won't keep us there. Let's get on to what we're doing to remain the leading independent Commodore magazine.

By now many of you have dropped in on *Ahoy!'s Port of Call,* our program on the PlayNET system. Each Thursday night at 10 p.m., Commodore users around the country gather to talk about their favorite subject. B.W. Behling chairs the weekly sessions, usually with one or more *Ahoy!* writers or editors as guests. You must be a PlayNET subscriber to attend. If you're not, the advertisements on pages 58 and 59 will tell you how to become one free with a subscription to *Ahoy!* If you don't have a modem, you'll find out on those same pages how to obtain one free!

Speaking of subscribing to *Ahoy!*, you'll note that the rates listed in our contents page indicia have gone up. But the insert card elsewhere in this issue reflects the old prices. Fact is, we printed hundreds of thousands of those cards before we knew our subscription prices would be rising. You can take advantage of our mistake by subscribing now at the old rate. We must emphasize the word "now."

And *now*—the contents of the February '86 issue of *Ahoy!*:

One of our most original construction projects to date, Don Lewis' *Turnkey 64* teaches you to build an AUTO-EXEC cartridge for instant execution of a specified program upon power-up. A program for use with the cartridge is also provided. (Turn to page 53.)

And while you're in a hardware hacking mood, Greg Granville's *Aligning the 1541 Disk Drive* will enable you to deal with the universal malady of Commodore users. Greg also provides a program for determining if your 1541 is out of kilter. (Turn to page 83.)

Arnie Katz and crew present not one but two complete software surveys in this month's *Entertainment Software Section*—of flight simulators and individual sports competitions. Also provided are full-length reviews of *The Fourth Protocol, Decision in the Desert, Lode Runner's Rescue, On Track Model Car Racing, Law of the West,* and *Dave Winfield's Batter Up!* (Turn to page 41.)

Orson Scott Card's *Creating Your Own Games* returns with an explanation of *Little Windows*, or how to incorporate multiple windows into your C-64 game screens. (Turn to page 18.)

You never know what Dale Rupert will cover next—and, frankly, neither do we until the *Rupert Report* lands on our desk each month. This time he turns your 64 into an Elementary Learning Machine—*ELMAC* for short. (Turn to page 34.)

In an expanded installment of *Commodore Roots*, Mark Andrews tutors assembly language students on *Memory Magic*—including how to copy your 64's built-in character set from ROM into RAM. Why, you ask? So that next month you can turn the character set into custom-designed text or graphics! (Turn to page 73.)

Beginning BASIC programmers are directed to this month's *Cadet's Column* for one of the more valuable lessons they'll ever learn—on structured programming. Additionally, Cheryl Peterson provides *Compuphobia*, a program you'll learn from and have fun with at the same time. (Turn to page 77.)

We're especially proud of this month's complement of ready-to-enter games—four of the finest ever to share an issue of *Ahoy!*:

Treasure Wheel by regular contributor Bob Blackmer simulates the TV game show of almost-the-same name. (Turn to page 31.)

Head to Head provides true two-player action as you and your opponent race to solve twin computer-generated mazes. (Turn to page 72.)

In *Arena* you wield a propellerlike spinner against fast-moving geometric shapes. (Turn to page 92.)

Crabfight combines two crustacea, a clam, and a pair of octopi in one of the wittiest games we've seen in some time. (Turn to page 69.)

There's much more inside, including *Reviews* of Cardco's *S'MORE* cartridge, InfoDesign's *Edge* series, and other hardware and software (turn to page 57); a mega-dose of frustration in *Commodares* (turn to page 77); *Tips Ahoy!*, providing the best programming hints and short routines available from the Commodore community (turn to page 87); and the *Scuttlebutt* on all the latest releases for Commodore computers (turn to page 8).

A number of readers have called or written to ask what the extent of *Ahoy!*'s Amiga coverage will be. Our answer: it depends on the extent of Amiga's success. We're far more interested in servicing the readers we have now than in running the risk of losing them by courting new ones—which is what would happen if we diluted our present Commodore 64, 128, and VIC 20 coverage to include the Amiga. While we will be reporting on Amiga releases in *Scuttlebutt*, we won't be publishing Amiga programs or articles until we're convinced that a reasonable portion of our audience and potential audience are using them. As we write these words in early December, it's clear that that's not yet the case.

Until it is, we hope you enjoy the undiluted Commodore 64, 128, and VIC 20 coverage in this issue of the leading independent Commodore magazine!

SCUTTLEBUTT

1541 CONTROLLER • CUSTOM CIRCUIT BOARDS • GAMES FROM FIRST STAR, SUBLOGIC, ELECTRONIC ARTS • C-128 BOOKS • INVOICER • FUNCTION KEY DEFINER • TAX PROGRAMS • MIDI SEQUENCER • DISK ORGANIZER • FREEWARE AND PD LIBRARIES • TALKING TERMINAL

SPOILS OF VICTORY

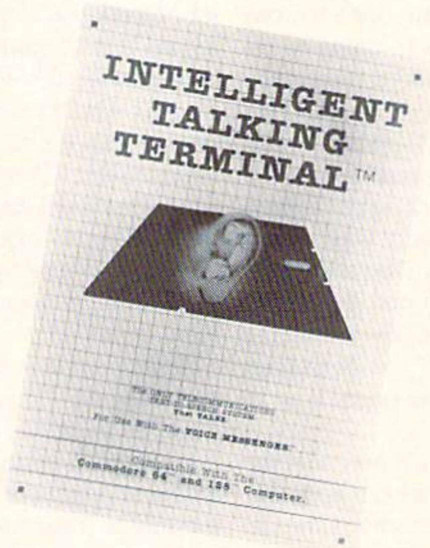
Victory Software is distributing its entire software line as freeware, encouraging user groups to copy and distribute the programs among members and non-members. Those who like what they get are asked to send a small donation to help perpetuate the service. Volumes I, II, and III, each containing 8-10 game and home utility programs, should be coming to a user group near you, or else may be purchased from Victory for $7.00 per disk ($5.00 without instructions).

Victory Software, 50 South Valley Road B-2, Paoli, PA 19301.

HOLY SCROLLER

Sunday School Master Writer holds 800 people records on a single disk, enabling church record keepers to input attendance and visit data and print basic people data, quarter roll sheets, birthday reports, address labels, and attendance reports. Attendance can be added up and averaged over consecutive Sundays. The documentation includes suggested forms to be used for communication between the church school office and the class. For the 64; $99.95.

UHL Research Associates, Inc., 7926 Berner St., Long Beach, CA 90808 (phone: 213-493-1955).



Receive data silently or verbally.
READER SERVICE NO. 201

TELECOM NEWS

Welwyn Currah's *Intelligent Talking Terminal* ($29.95), used in combination with the Voice Messenger (see December '85 *Ahoy!*), lets C-64 users receive modem-transmitted data verbally. Used alone, it serves as a silent telecommunications program that offers 80-column operation, capture and display of hi-res characters, redefinition of all keys to send text strings, change colors, or perform other functions, autologon, automatic formatting, and other features. The program is compatible with sequential word processors, and can be customized to suit the user's needs.

Welwyn Currah, 104 W. Fourth Street, Royal Oak, MI 48067 (phone: 617-353-1143).

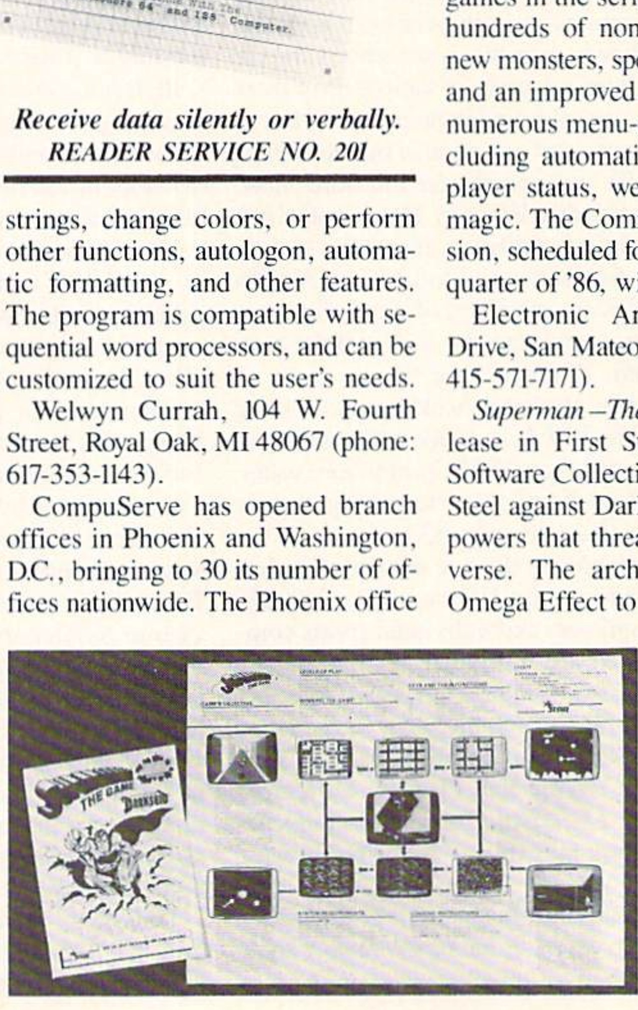
CompuServe has opened branch offices in Phoenix and Washington, D.C., bringing to 30 its number of offices nationwide. The Phoenix office is located at Suite E-104, 11221 N. 28th Drive, Phoenix, AZ 85029 (phone: 602-863-1404); the Washington office at Suite 700, 1655 N. Fort Meyer Drive, Arlington, VA (phone: 703-528-3393).

NEW GAME RELEASES

Ultimately, you knew there would be an *Ultima IV: Quest of the Avatar*, made available through Electronic Arts. The program, which fills both sides of two disks, incorporates a terrain 16 times larger than prior games in the series, interaction with hundreds of non-player characters, new monsters, spells, and challenges, and an improved user interface with numerous menu-driven sections, including automatic help features for player status, weapons, armor, and magic. The Commodore 64/128 version, scheduled for release in the first quarter of '86, will retail for $64.00.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403 (phone: 415-571-7171).

Superman–The Game, the first release in First Star's Super Powers Software Collection, pits the Man of Steel against Darkseid, a villain with powers that threaten the entire universe. The archfiend employs the Omega Effect to teleport citizens of



Superman–The Game *divides Metropolis into sections. Players must qualify to travel from screen to screen.*
READER SERVICE NO. 202

"IT DON'T GET NO BETTER THAN THIS"

EPYX

Summer Games	23.95
Summer Games II	23.95
Winter Games	23.95
Pitstop II	23.95
Eldolon	23.95
Rescue on Fractalus	23.95
Fast Load	23.95
Koronis Rift	23.95
Ball Blazer	23.95
Greatest Football Game	23.95
Multiplan 64/128	40.95
Progtool Kit	24.95

INFOCOM

Wishbringer	21.95
Enchanter	21.95
Zork 1-2-3	23.95 ea.
Suspect	21.95
Witness	21.95
Sorcerer	23.95
Cuthroats	21.95
Infidel	23.95
Spellbreaker	27.95
ALL INVISICLUES	6.00

ACCESS

Beachead II	21.95
Raid over Moscow	23.95

SIERRA

Wiztype	21.95
Champ Boxing	21.95
Stunt Flyer	16.95
Homeword	29.95
Homeword w speller	40.95
Ultima II	35.95

RIBBONS

MPS 803	6.95
SG10-15	2.35
Oki 82,92,83,93	2.35
Oki 182, 192	9.75
1526	6.50
Legend 880, 1080, 808	6.50
Epson FX, RX, MX, LX 80	4.50
Epson FX, RX, MX, 100	6.50
Epson 85	6.50
Epson 185	6.50
Brother 15,25,35	4.99
Juki 6000,6100	2.50
1525	6.75
Banana	6.75
MPS 801	6.75
Axiom Elite 5 LCD	7.99

MIN 6 Ribbons per order of any type.

BRODERBUND

Print Shop	27.95
P/S Library 1, 2,&3	16.95 ea.
Whistlers Bros.	18.95
Karateka	18.95
Music Shop	27.95
Bank St. Writer	29.97
Stealth	18.95

ACTIVISION

Slot Car Construction	21.95
Cross Ctry Road Race	19.95
Master of the Lamps	19.95
On Track Racing	19.95
Countdown to Shutdown	19.95
Gamemaker	24.95

BATTERIES INCLUDED

Home Pack	29.95
Consultant	36.95
Paper Clip 64/128	36.95
P/C w/Spell 64/128	49.95

MICRO LEAGUE

Microleague Baseball	29.95
Managers Disk	29.95
Team Disks	14.95

MICROPROSE

Crusade in Europe	24.95
F-15 Strike Eagle	21.95
Kennedy Approach	21.95
Silent Service	21.95
Acro Jet	21.95
Gunship	Call
Decision in Desert	24.95

TIMEWORKS

Data Manager II	29.95
Data Mgr. II 128	42.95
Electronic Checkbook	16.95
Money Manager	16.95
Evelyn Wood Reading	29.95
SwiftCalc 128 w/Sideways	42.95
Word Writer 128	42.95
Swift Calc w/Sideways 64	29.95
Sideways	19.95
Word Writer w/Speller	29.95

ACCOLADE

Flight Night	19.95
Dam Busters	19.95
Hard Ball	19.95
Law of the West	19.95

DISKS
Guaranteed Error Free

Generic No Label Pack of 25 w/sleeves	18.75
Maxell MD 1D	12.95
Drive Head Cleaning Kit	11.95

STRATEGIC SIM

Carrier Force	35.95
Computer Baseball	23.95
Computer Quarterback	23.95
Field of Fire	23.95
Kampgreuf	35.95
Mech Brigade	35.95
Operation Mkt. Garden	29.95
Germany 1985	35.95

MISC.

Spy vs Spy	18.95
Spy vs Spy II	18.95
Tapper	27.95
Doodle	23.95
States and Traits	27.95
Typing Tutor III	24.95
Body Transparent	27.95
I am the 64	18.95
CompuServe Starter	22.95
Superbase 64 V.2	46.95
Superbase 128	56.95
Practicalc 64	18.95
Practifile 64	18.95
Tax Advantage '85	Call
Sargon III	29.95
Koala Printer	16.95
Super Script	Call
Super Bowl Sunday	19.95
Kung FU	19.95
Karate Champ	21.95
C-64 Koala Pads	42.95
Flight Sim. II	29.95
Fl. Sim. Scenery	13.95
Intl. Hockey	16.95
Halley Project	23.95
Ultima III	35.95
Ultima IV	39.95
Superman	19.95

COMPLETE LINE OF APPLE & IBM SOFTWARE

ACCESSORIES & DISKS

Disk Notchers	6.50
Disk Mailers	0.49 ea. (min. 10)
Drive and Printer Cables (specify length)	1.20 per foot (6 ft. min.)

AMERICAN INTERNATIONAL COMPUTER PRODUCTS

P.O. Box 1758, STATEN ISLAND, NY 10314
PHONE ORDERS 718-351-1864

TRY OUR GREAT EXPRESS SERVICE

****SPECIAL INTRODUCTORY OFFER****

FREE 1541 Disk Drive Cleaning Program With Every Order

Ordering Info: In stock items shipped within 24 hrs. for prepaid orders (cert. chk. - visa - m/c). C.O.D. orders accepted please add $4.00 must be cash or cert. chk. only. NYS residents add 8¼% sales tax. Visa & Master Card add 4%. All orders add 3.50 shipping and handling. International Orders add $5.00 $ S & H. School inquiries invited. Personal checks allow 2 wks. for clearance. Availability and pricing subject to change without notice. Orders over $150.00 receive free shipping in continental U.S. Make all checks payable to: American International Computer Products. Defective Mdse: Handled by authorized Warranty Service Center in your area. Hardware special subject to availability.

SUPER HARDWARE PACKAGE

C-128 Computer
1571 Disk Drive
Super Epson Compatible
NLQ PRINTER
$685.00 S & H Included
PRINTER INTERFACE ADD. $44.95

Metropolis to his underground lair where he can try to wrest the secret of the Anti-Life formula from them. Against him, Superman pits his powers of flight, super strength, heat vision, and super breath. Players can assume the roles of hero and villain and compete against each other or play solo against the artificial intelligence-aided computer. For the C-64; $29.95.

If you purchase the program and send in the proof-of-purchase along with the coupon available at your dealer and $2.99, you'll receive a Super Powers Digital Watch valued at more than $10.00.

First Star Software Inc., 18 East 41st Street, New York, NY 10017 (phone: 212-532-4666).

Battle for Midway puts the C-64 user in command of the United States Pacific Fleet just following the December 7, 1941 attack on Pearl Harbor. Previously available in Europe, it is now available in the US from BCI for $9.99.

BCI Software, P.O. Box 730, Ringwood, NJ 07456 (phone: 201-835-7300).



SubLOGIC's 68000-based software development team takes time out from their work on third-generation flight simulators to pose for the camera. Bless you boys.

Epyx Sports Preview enables C-64 owners to sample *The World's Greatest Baseball Game, Winter Games, The World's Greatest Football Game, Summer Games,* and *Summer Games II* for $3.50 ($4.00 in Canada). A $3.50 rebate is offered for any Epyx sports game purchased before February 15, 1986.

Epyx Computer Software, 1043 Kiel Court, Sunnyvale, CA 94089 (phone: 408-745-0700).

SubLOGIC has enhanced its *Flight Simulator* (see *The Digital Blue Yonder,* page 41) for 68000-based computers, including the Amiga. Improvements include the choice of flying a Cessna 182 or a Lear Jet (by mouse or keyboard), pull-down menus for setting time or season and cloud layers by pointing a mouse-controlled cursor, multiple 3D windows for watching two or more views simultaneously, and control tower and spotter plane views in addition to the original nine. Each window can be individually adjusted for the best viewing angle, from wide-angle (1X) to super-telephoto (4000X).

SubLOGIC Corporation, 713 Edgebrook Drive, Champaign, IL 61820 (phone: 217-359-8482; orders 800-637-4983 except in IL, AK, and HI).

PD SOFTWARE

64 GOLD, a library of public domain Commodore 64 programs encompassing games, word processing, extended utilities, communications, and education has been made available for $5.00 each by the Blackship Group. A free catalog can be obtained by writing 64 GOLD, 3219 Folsom Street, San Francisco, CA 94110 (phone: 800-431-6249; in CA 415-550-0512).

BOOKS

The *Command Post* reprint booklet combines all the material that ran in *Commander* magazine's Commodore ham radio communications column of the same name from September 1983 through June 1984. Besides teaching basic interfacing techniques for amateur radio applications and other control situations and reviewing hardware and software, program listings are provided for morse code send and receive, radio-teletype send and receive, duplicate checking and program exchange over the air. Price is $9.95 plus $2.50 shipping in US and Canada; overseas orders plus $5.00 (full amount by international money order).

QSKY Publishing, P.O. Box 3042, Springfield, IL 62708.

More available details on *C-128 Internals* ($19.95), preannounced in November's *Scuttlebutt:* the volume includes information on using interrupts, assembly language programming and Kernal routines, the Z-80 processor and the boot ROM, peripherals and the ports, programming for sound and music, the various graphics modes, and the memory management unit, the input/output ports, and using the 80-column chip for 640 X 200 point resolution, smooth scrolling, copying blocks in screen memory, character length and width management, and getting more than 25 lines on the screen.

Abacus Software, P.O. Box 7211, Grand Rapids, MI 49510 (phone: 616-241-5510).

The Commodore 128 Mode: An Inside View explores the computer's 16 memory banks, with information on bank switching, programming with the built-in monitor, BASIC 7.0 tokens and graphic and sound commands, and disk storage on the 1541 and 1571. Also included are demonstration programs and an instructive memory map.

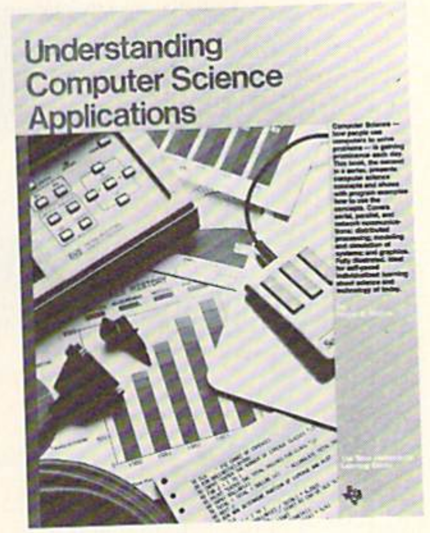
Microcomscribe, 8982 Stimson Ct., San Diego, CA 92129 (phone: 619-484-3884 or 578-4588).

Locating Computer Programs ($17.00) guides the micro user to over 300 sources of software information, including directories, books of programs, reviews, databases, and national software-sharing organizations.

Garland Publishing, 136 Madison Avenue, New York, NY 10016 (phone: 212-686-7492).

Understanding Computer Science Applications ($14.95) provides a self-paced course on how computers are used to solve problems. Covered are serial, parallel, and network communications, distributed processing, modeling and simulation of systems, graphics, and future applications. Each chapter ends with a summary and quiz.

Texas Instruments Incorporated, P.O. Box 225474, MS/8218, Dallas, TX 75265 (phone: 214-997-3955).



A self-paced, textbook-style course.
READER SERVICE NO. 204

EDUCATIONAL PROGRAMS

Tiger's Tales teaches kindergarteners through second graders reading comprehension, vocabulary, and problem-solving skills as they join tabby on a series of adventures, helping to decide how each story will proceed. The Words section that accompanies each of the five stories lets children match up unfamiliar words used in the stories with pictures. The C-64 program can be used in conjunction with Koala's Muppet Learning Keys. Price of the disk plus one backup and teacher's guide is $59; lab package (10 disks plus teacher's guide) is $177.

Sunburst Communications, Inc., 39 Washington Ave., Pleasantville, NY 10570 (phone: 800-431-1934; in NY, AK, and Canada call 914-769-5030 collect for orders).

PLANNING CODE

Zip helps managers plan projects by selecting the best mix of objectives that can be accomplished with the resources available, asking "what if" questions and determining their effect on a project, defining objectives and the activities and resources needed to achieve them, tracking and managing project costs, and more. Price of the complete C-64 package is $99.95 plus $10.00 shipping, $29.95 plus $10.00 for the documentation only, or $79.95 plus $5.00 for the disks only.

Lawco Ltd., P.O. Box 2009, Manteca, CA 95336 (phone: 209-239-6006).

PROBLEM PROGRAM

Everyone you know butts into your affairs anyway. So why not let your 64 do it? *Creative Problem Solving* ($29.95) evaluates personal, financial, and business problems by weighing user-entered input and producing a printed analysis of the factors involved. The output shows the best solution to the problem, but can also be utilized as a list of pros and cons. A printer is necessary.

Harvsoft, Box 725, Kenmore, NY 14217 (phone: 716-877-3510).

MOVIE MAKER REMAKE

Electronic Arts has reintroduced *Movie Maker* (formerly available

from Reston Publishing), enhanced with over 100 new drawings and three animated movies by Gahan Wilson. These and the other seven built-in movies can be customized or studied as tutorials. The program permits users to create animated tracks, or characters, of up to 300 frames in length, and to combine up to six such tracks in a single animated sequence. The sequences can be chained together for automatic playback, or a VCR can be used to combine them into feature-length animated productions. A library of built-in tones and sound effects is included, allowing the creation of different sound tracks for each movie. For the C-64; $32.95.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403 (phone: 415-571-7171).

ANTI-STATIC PRODUCTS

The *Static Buster* line of static control products has been expanded to include Anti-Stat Cleaner (reduces glare, static, and dust buildup while leaving an invisible conductive film) and Anti-Stat Wipes, both designed to work with the Static Potential Equalizer.

Lamb's Information Systems, 301 N. Main St., Pueblo, CO 81003 (phone: 1-800-STATIC-1).

NEW DISK MARKERS

Red, blue, and green Diskribe markers have been added to the original gold and silver, allowing users to color-code a broader range of disks. The markers are designed to write directly on the disk jacket. Price is $2.50 each.

Sanford Corporation, 2740 Washington Blvd., Bellwood, IL 60104.

MACH 128

Designed for use with the C-128 or C-64, the *Mach 128 Enhancement Package* ($49.95) will speed 1541 or 1571 loads by up to five times, provide 40- and 80-column screen dumps, and make DOS enhancements available on the C-64 and in all three 128 modes: 40-column, 80-column, and C-64.

A mode selection switch on the cartridge allows the user to choose either native 128 or C-64 emulation mode. A reset switch allows for warm starts.



Static Buster Anti-Stat Wipes are treated, lint-free cloths used to apply Static Buster Anti-Stat Cleaner or for everyday dusting. Both work with the Static Potential Equalizer as a total system. READER SERVICE NO. 205

The cartridge plugs into the expansion port of either machine and may be left in at all times. Upon power up, the cartridge senses whether a 1541 or 1571 drive is connected to the system and which mode the computer is in, then engages or disengages the drive speed routines dependent on the hardware configuration.

Included in the package is a disk containing a BASIC +4K program, two monitors, a disk organizers and catalog program, and other utilities.

Access Software, Inc., 2561 South 1560 West, Woods Cross, UT 84087 (phone: 801-298-9077).

DISKORGANIZER

Diskorganizer ($29.95) allows C-64 users to sort and rearrange the disk directory of any unprotected disk to meet specifications, and write the new directory to the disk. A screen editor permits changing the header, scratching, scratchproofing, and renaming files, writing files to another disk, marking off sections of the disk for easy reading and independent sorting, positioning individual files anywhere in the directory, and printing copies of revised directories.

The G.A.S.S. Company, 970 Copeland, North Bay, ONT P1B 3E4 (phone: 705-474-9602).

SENTENCE GENERATOR

Sort of a computerized Mad-Lib book, *Creative Writer* ($29.95) generates an infinite number of random, gramatically correct sentences. Suggested uses are simulation of famous authors and tailor-made abuse for politicians and friends.

The G.A.S.S. Company, 970 Copeland, North Bay, ONT P1B 3E4 (phone: 705-474-9602).

TAX PROGRAMS

Two programs for taxpayers from Northland Accounting:

Bookkeeper's Aid complies with all IRS record keeping requirements in keeping the small business tax records straight. The program is menu-driven, with all data input assisted by screen prompts. 300 user-defined account classifications are available, and all sorting and calculating is performed independent of the disk or tape drive. The program will work with most serial printers to generate a complete report of all records by record number and sorted by account. On disk or tape for the VIC 20 with 16K, the C-64, or the Plus/4; $34.95.

Revised once again to incorporate all the latest tax laws, *Taxaid* will prepare any IRS form 1040 and print the results to screen or paper. The user can make changes and revisions at any time during data entry. Calculations are automatic and all tax tables, including income tax averaging, are built-in. On disk or tape for the VIC 20 with 16K, the C-64, or the Plus/4; $39.95.

Northland Accounting, Inc., 606 Second Avenue, Two Harbors, MN 55616 (phone: 218-834-3600).

CUSTOM BOARDS

New custom circuit boards for the C-64 and 128 from Micro R & D:

The MW-611 universal I/O board plugs into the C-64/128 expansion port, providing for 16 channels of analog inupt and from 14 to 16 dis-

crete outputs as well as an additional analog output. All outputs are programmable through BASIC, individually or in groups. Diode suppression is provided to protext the output from inductive kickbacks. Other features include an EPROM socket for custom programs, A/D and D/A converters, and a 16-pin footprint for experimental circuits.

The MW-150 is a terminal breakout board for any analog/digital conversion board, featuring circuitry for using a thermocouple amplifier with cold junction compensation to produce an amplified output signal from a thermocouple input. The device senses the junction temperature on the board using a temperature-sensitive amplifier and provides internal compensation. Also included are circuits for RC noise filtering and scaling voltages down using a voltage divider circuit.

Micro R&D has enhanced their MW-350 Graphics Printer Interface (see February '85 *Ahoy!*) to include a 4K buffer and a revised operating ROM that provides for switch settings for eight popular printer brands.

Micro R&D Inc., 3333 S. Wadsworth A-104, Lakewood, CO 80227 (phone: 303-985-4077 or 1473).

GRAPHICS DISKS

Broderbund and Springboard have enhanced their popular C-64 graphics programs with new supplementary art disks.

The Print Shop Graphics Library Disk 3 ($24.95) provides 120 new designs, symbols, and pictures, including international symbols, dozens of business icons, 15 new Christmas graphics, and graphics representing myth and fantasy, the seasons of the year, animals, and miscellaneous designs.

Broderbund Software, Inc., 17 Paul Drive, San Rafael, CA 94903-2101 (phone: 415-479-1170).

The Clip Art Collection, Volume 1 ($29.95) adds 600 pieces of art to the 600 original pieces provided with *The Newsroom*, Springboard's $49.95 newspaper-making program for the Commodore 64 and 128.

Springboard Software, Inc., 7808 Creekridge Circle, Minneapolis, MN 55435 (phone: 612-944-3915).

1541 CONTROL

The 1541 Control ($34.95) mounts on top of the drive, allowing the user to change the device number, increase disk storage by 100%, and reset the computer, disk drive, or entire system. Installation is facilitated by an illustrated manual.

Micro Fabrication Systems, P.O. Box 1114, Delaware, OH 43015.

INVOICING PROGRAM

The Invoicer ($59.95), which lets C-64 owners produce bills on blank paper or preprinted forms, features batch or real time processing, user- defined smart keys, automatic letterhead formatting, sales tax calculation, line extension and totaling, built-in audit controls, and daily sales reporting with optional user-defined item subtotals. Tax, taxable sales, and nontaxable sales are automatically tracked. Statistical data such as average invoice amount can also be generated.

MiccaSoft, 406 Windsor Lane, New Braunfels, TX 78130 (phone: 512-629-4341).

FUNCTION KEY DEFINER

Screen Dump, Etc. ($24.95) defines the C-64's eight function keys for different tasks. Besides enabling you to print a C-64 screen of any resolution or combination of resolutions to a dot-addressable printer, the program will let you save screens to and load screens from disk, display screen memory allocations, perform decimal/hex conversion, and get HELP. One key is user-definable. All function key routines can be performed at any time, even during program execution.

IRQ, Inc., P.O. Box 457, St. Charles, MO 63302.

MIDI SEQUENCER

Two C-64 MIDI products from Dr. T's Music Software (requiring a Model-T, Sequential 242 or 64-compatible, Passport, or Passport-compatible MIDI interface):

The *Keyboard Controlled Sequencer* ($125.00) can be used as a 16-track tape recorder, capable of recording and playing back on any or all of the 16 MIDI channels. Included is a flexible set of options for structuring music, real-time control of up to 35 independent sequences from the computer keyboard, and an editing system that displays up to 22 notes onscreen at a time and lets the composer change the timing, dynamics, pitch, or MIDI channel. Music can be recorded in three ways: in real-time from the synthesizer keyboard, one note at a time from the computer keyboard, or as sequences of evenly spaced notes entered from the synthesizer keyboard using the STEP-TIME function.



1541 Control: change device #, reset.
READER SERVICE NO. 206

The *DX Patch Librarian* ($75.00), originally intended for creating and storing patches for Yamaha DX7 synthesizers, has been modified to permit programming of TX and 8-16 expansion modules without a DX7. It allows the C-64 user to attach a keyboard controller or external sequencer to the MIDI-in of the computer to easily hear his TX sounds.

Dr. T's Music Software, 66 Louise Road, Chestnut Hill, MA 02167 (phone: 617-244-6954).

TALKING TYPING TUTOR

The latest of the Tymac Talkies, software-generated talking games, *Type-Snyper* teaches typing through 16 different exercises, with lessons ranging from proper finger placement to increasing speed and accuracy. Gameplay consists of an attack helicopter dropping bundles of words and letters that must be "destroyed" by typing them before they reach the city. Exercises are automatically adjusted to emphasize areas needing extra help. For the 64; $39.95.

Tymac Controls Corporation, 127 Main Street, Franklin, NJ 07416 (phone: 201-827-4050).

COMMODORE PRINTER

Hattori Seiko's SP-1000VC dot matrix printer ($270.00) offers Commodore compatibility, bidirectional printing with logic-seeking, adjustable tractor and friction feed, a variety of print character sets, automatic paper loading, and right/left margin set functions.

Hattori Seiko Computer Peripherals, 1111 MacArthur Boulevard, Mahwah, NJ 07430 (phone: 201-529-5730).

BASIC TOOLKIT

The *BASIC Toolkit* provides more than 100 new BASIC commands, allowing Commodore 64 and 128 programmers to add high-speed assembly language operations to their application programs. In addition to enhancing BASIC, the *Toolkit* boosts the Commodore operating system, with features like a directory listing command, straightforward disk command syntax, and a LOAD/SAVE function for graphics, sprites, and character sets. Built in is the Vorpal Fast Loader, Epyx's own rapid disk-loading enhancement for the 1541 and 1571 disk drives.

Users may print disk catalogs, text files, and program listings in BASIC. Commands are provided for sophisticated printing utilities, as well as for incorporating joystick functions into programs.

Special graphics commands facilitate the development of application programs for home or business-use graphics. Graphics like those found in many commercially available games are made possible for users with artistic talent.

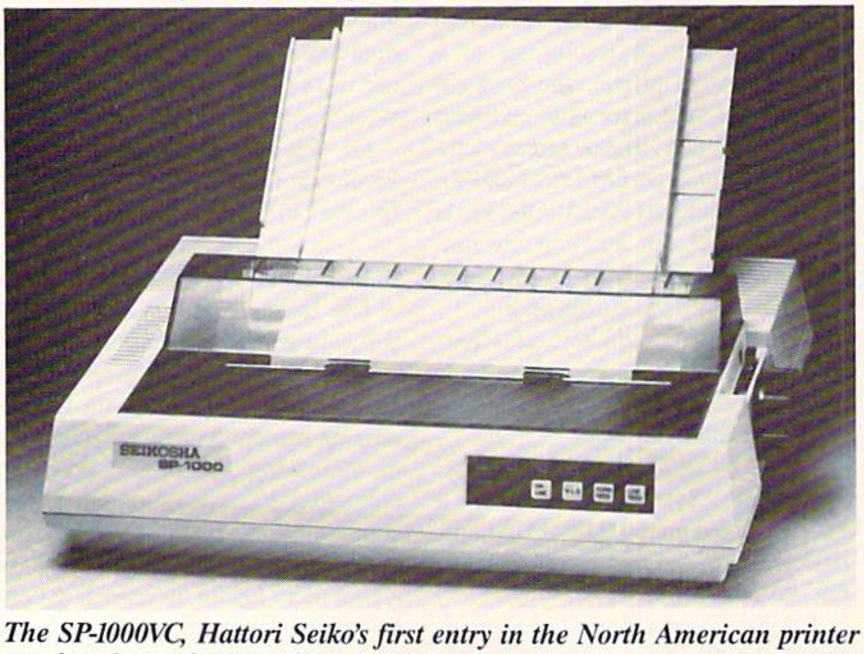
Each command is documented in a *Toolkit* manual that includes demo and utility programs plus a command reference card. Retail price of the program will range from $34.95 to $44.95.

Epyx, Inc., 1043 Kiel Court, Sunnyvale, CA 94089 (phone: 408-745-0700).

GRAPHICS INTEGRATOR

From the makers of *Flexidraw*, the *Graphics Integrator* will convert picture files from one graphics package for use in another. Conversion can be made from hi-res to hi-res, hi-res to *Flexidraw* format, multicolor to multicolor, or multicolor to hi-res (with some conversion loss). Word processor compatible output with packages having an external file input capability allows the integration of text and pictures. The program can also be used to create self-running slide shows in hi-res, generate stand-alone picture files that can be entered as BASIC, and print pictures from graphics programs not currently supporting printer drivers.

Inkwell Systems, P.O. Box 85152 MB290, San Diego, CA 92138 (phone: 619-268-8792).



The SP-1000VC, Hattori Seiko's first entry in the North American printer market, has a built-in Commodore interface and a two-year warranty.

PRINTER ACCESSORIES

Via its two easy-to-install ROM chips, the Special Edition Kit for the Star Micronics SG-10C printer eliminates hardware selection of print functions and offers software accessibility of print modes such as near letter quality, italic, super- and subscript, density bit image graphics, and underlining. Star distributors will offer the kit free of charge to all SG-10C owners and purchasers.

Star Micronics Inc., 200 Park Avenue, Suite 3510, New York, NY 10166 (phone: 212-986-6770).

A Plug 'N Print Kit designed for the Amiga will allow users to print all 4096 of the Amiga's screen colors with the Okimate 20 printer. Included is interface module, data cable, color and black ribbons, and paper. For a limited time, kits will include a disk that demonstrates the picture capabilities of the Amiga's 512K model, as well as the Okimate's text font and the Amiga's graphics font capabilities. The kit will retail for $99. (The Okimate 20 sells for $169.)

Okidata, 532 Fellowship Road, Mt. Laurel, NJ 08054 (phone: 609-235-2600).

SUNCOM PRODUCTS

Two new computer accessories from Suncom:

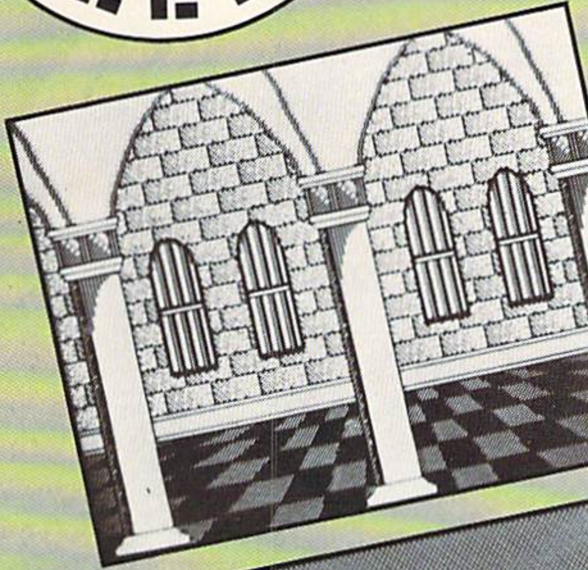
The Notcher ($9.95) punches a hole on jackets of single-sided diskettes in the place occupied by the write-protect notch on a double-sided disk. While not intended to render the disk's second side useful for primary data storage, it will provide the user with a means of manufacturing backup or duplicate copies of important data disks.

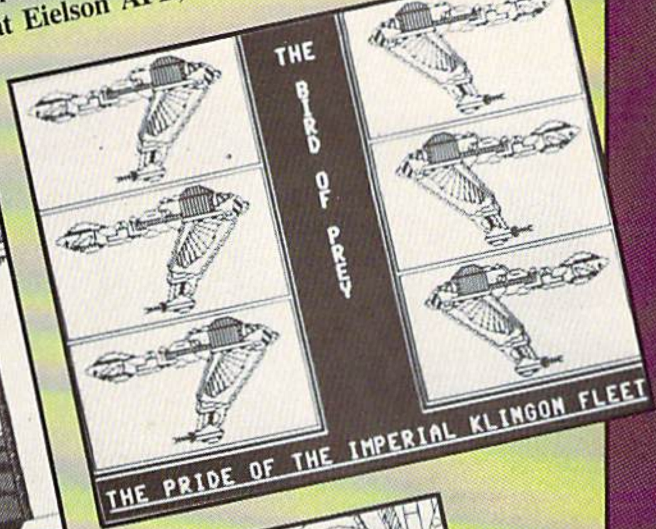
The Wet System Disk Head Cleaner ($7.95) removes particle matter, smoke residue, dust, and corrosion that build up on disk heads after extended periods of use.

Suncom Incorporated, 260 Holbrook Drive, Wheeling, IL 60090 (phone: 312-459-8000).

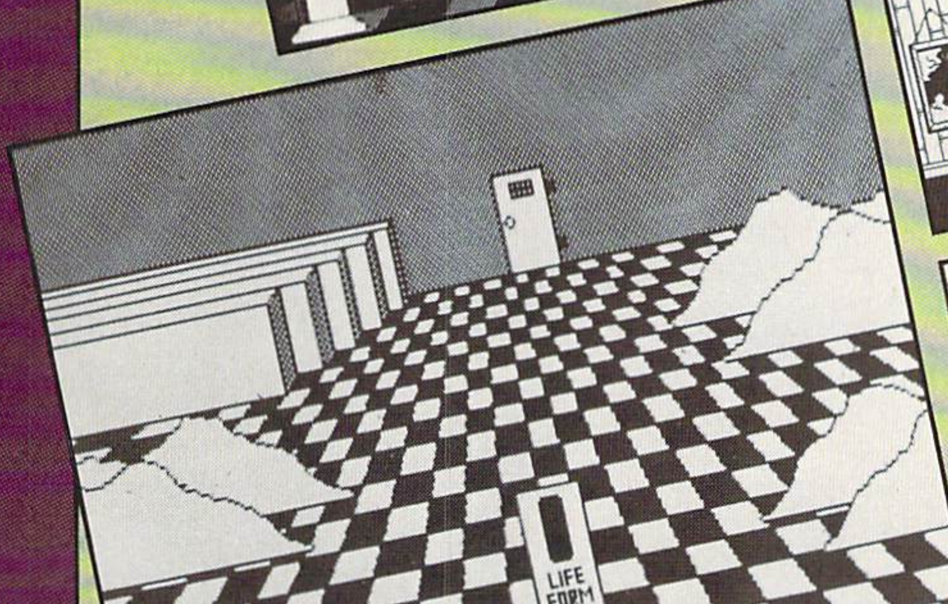
ART GALLERY

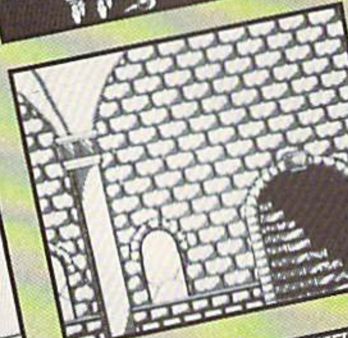
Steven Distasio's (Valhalla, NY) images in last month's *Art Gallery* were so striking that we've devoted this month's entire installment to high-resolution black and white drawings—five of them by Mr. Distasio, all rendered with *DOODLE!* and, like last month's contributions, all demonstrating the maximum resolution of the 64's 320 X 200 pixel hi-res mode: *Dungeon #1*, *Bird of Prey* (top), *Life Form*, *Hobbit House*, and *Dungeon #2* (below). In the lower right hand corner is another hi-res *DOODLE!*, *Hooked* by Michael C. Hulin (stationed at Eielson AFB, Alaska).















New toll free number:
1-800-541-1541

KEYMASTER

The easiest-to-use, most-powerful disk backup program ever written for the 64.

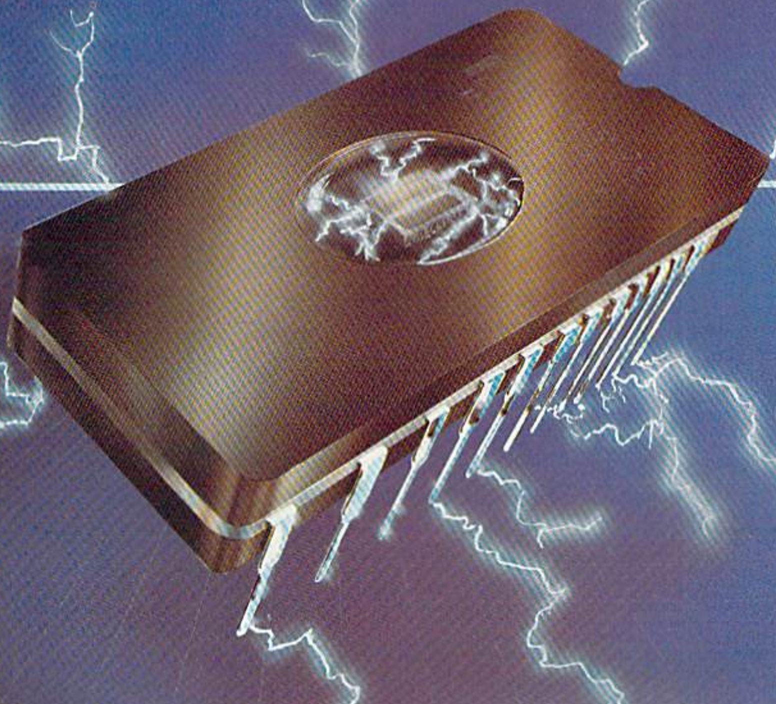
With one 1541 drive:
- **FAST** FILE COPIER — up to 5 times faster!
- **FAST** full protected — disk GCR nibble copier
- **FAST** full unprotected — disk copier
- **KEYMASTER** protected — disk unlocking kit**
- **FAST** 10 second disk formatter

With two 1541 drives (NO disk swapping required during copying):
- **FAST** file copier — up to 6 times faster!
- **FAST** full protected — disk GCR nibble copier
- **FAST** full unprotected — disk copier

** **KEYMASTER** unlocks protected disks by **ELIMINATING** the protection scheme originally written on the disk.
** Once one of KEYMASTER's KEYS has unlocked a disk, the program can generally be **FILE COPIED** onto another disk (also making the program compatible with non-1541 drives). If it is not file copyable, it can be fast nibble copied.
** **50 KEYS** for popular programs will be provid ded on the first KEYMASTER disk.
** Frequent updates of 25 to 50 additional KEYS will be available (no original disk return required for updates!).

ALL OF THIS AT THE RIGHT PRICE! ONLY —
$29.95 for KEYMASTER
$10.00 for update disks

KEYMASTER created by Jim Drew.



New toll free number:
1-800-541-1541

MAGNUM LOAD BLOWS THE OTHERS AWAY!

ATTENTION 1541 AND MSD OWNERS!

- MAGNUM LOAD will LOAD and VERIFY programs up to **6 TIMES FASTER** than before. It is a new replacement KERNAL (operating system) ROM chip for your Commodore 64 or 128 computer.
- A high-speed loader, high-speed verify, disk drive no-head-rattle routine (1541 drive) and a BASIC UNNEW ROUTINE (MSD drive) have replaced the Datasette tape routines.
- The first program on a disk may be loaded and run by pressing only two keys (SHIFT/RUN).
- Unlike the older, cartridge-style fast loaders, no ports are tied up at the rear of your computer, the screen is not blanked during loading and there is no wear-and-tear on the game port.
- For maximum convenience and performance, MAGNUM LOAD is installed DIRECTLY in the circuit board. Generally, a socket has already been provided to make the operation easy, but occasionally some soldering may be required.
- Now you can give your 1541 disk drive "1571 speed."

The chart below compares ACTUAL MEASURED loading times.

PROGRAM	STAR DOS	REGULAR LOAD	MACH 5	FAST LOAD	MAGNUM LOAD
Pitstop II (EPYX)	?	144 sec	43 sec	41 sec	31 sec
Music Shop (BRODERBUND)	?	105 sec	105 sec*	105 sec*	21 sec
Hitchhiker's Guide to the Galaxy (INFOCOM)	?	70 sec	70 sec*	**	68 sec*
On-field Football (GAMESTAR)	?	159 sec	66 sec	63 sec	56 sec
EASY FINANCE I (COMMODORE)	?	58 sec	13 sec	13 sec	11 sec

* = Will not fast load — defaulted back to regular load.
** = *Failed to load at all.*

FIVE VERSIONS OF MAGNUM LOAD ARE AVAILABLE TO FIT ANY NEED.

VERSION	APPLICATION	PRICE
A	C64 computer and one 1541 drive	$24.95
B	C64 and one MSD drive (single or dual)	$24.95
C	C64, one 1541 drive AND one MSD drive (switchable)	$29.95
D	C128 computer and one 1541 drive***	$24.95
E	C128 and one MSD drive (single or dual)***	$24.95

*** = Fully compatible with 128 & CP/M modes. Cartridge versions MUST be UNPLUGGED to access 128 & CP/M modes!

ORDER THE VERSION YOU NEED NOW!

MAGNUM LOAD created by Jim Drew

C64, C128, 1541, 1571 and Datasette TM Commodore; MSD TM Micro Systems Development MACH 5 TM ACCESS; FAST LOAD TM EPYX; STAR DOS TM Star Point

THE SHADOW $89.95

Shadow is a new and revolutionary piece of hardware that is used to duplicate even the most protected software. Fitting inside the disk drive (no soldering required), SHADOW takes complete control of all functions giving near 100% copies.

Being the best utility available today, it will even copy the other copy programs.

Because of the Shadow's unique abilities, we feel DOS protection is a thing of the past.



*HACKER PACKAGE $39.95

Shadow a disk while it loads, then read an exact list of:

- Track, sector, ID, check sum, drive status
- High and low track limits
- Density use on each track
- Half tracks that are used
- Command recorder shows commands that were sent to 1541 while program was loading
- RAM recorder records custom DOS

Shadow-scan any disk, then read exact list of:

- Valid tracks, half tracks, partial tracks and segments
- Sync mark link, header block links and data block links
- Track to track synchronization

Exclusive snap shot recorder will give you an exact copy of the 1541 RAM and can be viewed, saved or printed. Plus many more features included.

*Requires Shadow

*GT PACKAGE $44.95

Highly sophisticated and integrated piece of hardware that turns you 1541 into something you've always wanted.

- Track and sector display
- Drive reset switch
- Device number change
- Half track indicator
- Abnormal bit density indicator
- Shadow on-off indicator

The Shadow display will give you an accurate display of precisely what track you are accessing during a normal load even if the program does a read past track 35.

*Requires Shadow



Order by phone 24 hrs./7 days or send cashier's check/money order payable to Megasoft. Visa, MasterCard include card number and expiration date. Add $3.50 shipping/handling for continental U.S., $5.50 for UPS air. CODs add $7.50, Canada add $10.00. Other foreign orders add $15.00 and remit certified U.S. funds only. Distributors invited and supported.

MegaSoft LTD

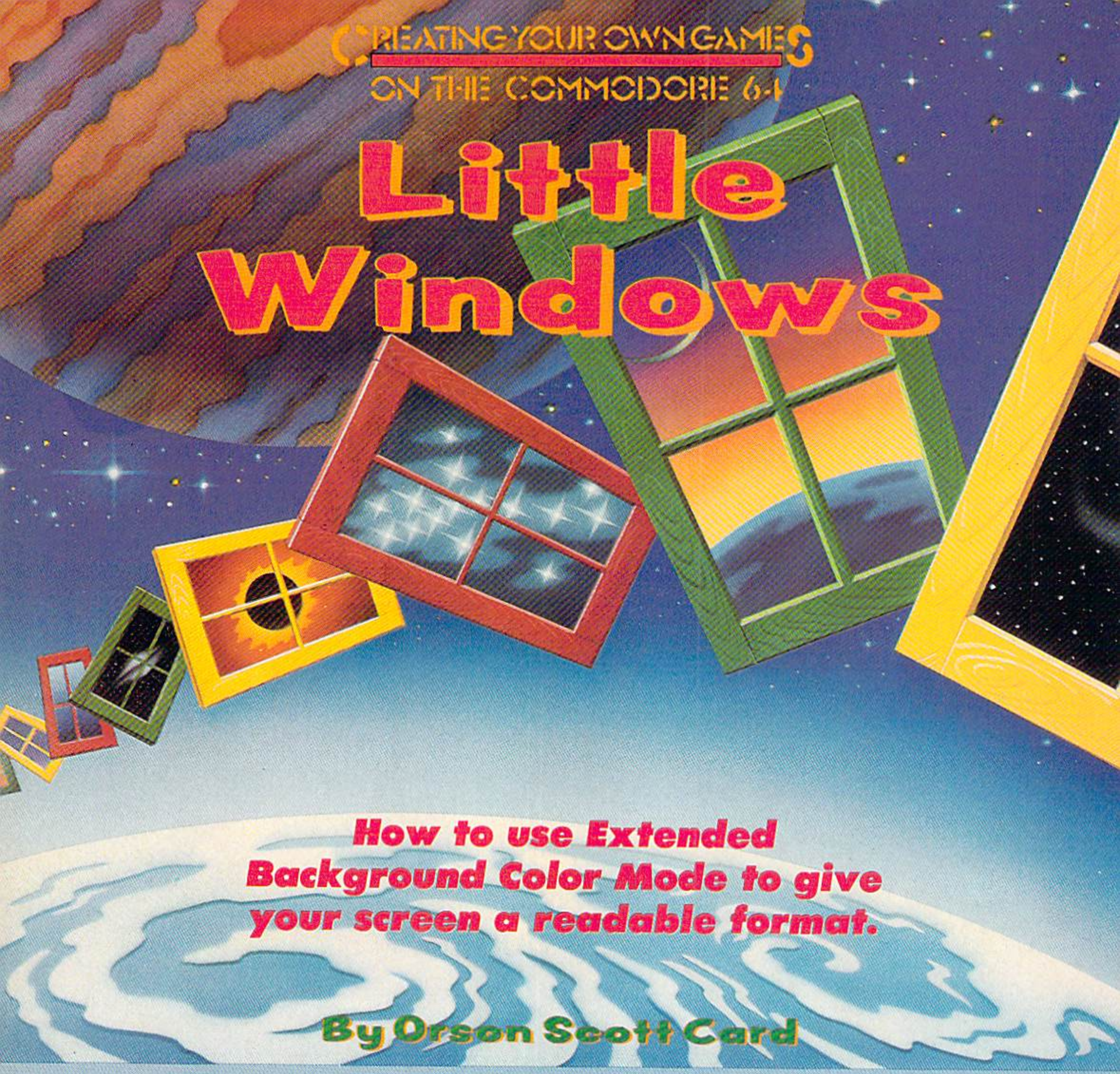
P.O. Box 1080 • Battle Ground, Washington 98604
1-800-541-1541
Canadian/Foreign Orders Call
(206) 687-5205



Reader Service No. 200

CREATING YOUR OWN GAMES ON THE COMMODORE 64

Little Windows



How to use Extended Background Color Mode to give your screen a readable format.

By Orson Scott Card

If the game you're creating relies heavily on text, the screen can get pretty boring. Of course, Infocom has shown us that if the story of the game is interesting enough, even plain vanilla text-only screens can keep our interest.

Remember, though, that the screen display isn't designed merely for entertainment. It is also your chief tool for *communicating* with the player.

By carefully designing the screen format, you can enhance the screen's ability to convey information clearly and quickly. If it also makes the screen more interesting—well, who can complain about that?

DEFINING WINDOWS

The most obvious way to make the screen more interesting is to change the foreground and background colors—and get rid of that boring light blue on dark blue. The foreground color (the color of the letters) can be changed by PRINTing one of the color-change characters (QUOTE, then SHIFT-1 through SHIFT-8 or COMMODORE-1 through COMMODORE-8). The background color is changed by POKEing a value from 0 to 15 into location 53281. And to change the border color, POKE a number from 0 to 15 into location 53280.

Another way to snazz up the display is to design a new character font, so we don't have to look at the same old letter shapes. We've gone through that process in detail in earlier columns—a year ago, if I remember correctly.

The trouble with both these changes is that as often as not they *interfere* with communication. Most char-



acter fonts don't improve the clarity of the letters; nine times out of ten, they make the letters even harder to read, which very quickly becomes far more annoying than the boredom of the regular character set.

And changing the colors is interesting for the first few moments—but there aren't many foreground-background color combinations that have the right contrast to be readable for the long hours that the player will want to play your game.

What will make your screen more interesting *and* improve its usefulness is to divide the screen into "windows" —sections of the screen used for the same purpose all through the game.

For instance, if you were creating an adventure game, you might reserve a small area—a single line at the bottom, for instance—that constantly displayed the elapsed time, the character's strength and health, and how much electricity his flashlight had left. The player could glance at that screen line any time he wanted the information, without having to use up a command finding it out.

You could use another window to give the name of the room he is in and any other characters who are in there with him. A third window could contain a constant list of what he's carrying. The fourth—and largest—window would then contain the traditional command-and-response information that all adventures seem to have.

The result would be a very pleasing visual design: four distinct screen areas, each used to convey a particular kind of information, so the player could look at the same place every time.

TOM CUSHWA

MAGNUM LOAD

Programed by Jim Drew

MAGNUM LOAD is a new replacement KERNAL (operating system) ROM chip for your Commodor 65 or 128 computer that will load and verify programs up to 6 times faster than before. The tape routines have been removed from the old chip and in their place have been put a high-speed loader, high-speed verify, and disk driver no-head-rattle routine. Unlike the older cartridge fast loaders, no ports are tied up at the rear of the computer, the screen is not blanked during loading and there is no wear-and-tear on the expansion port. For maximum convenience and performance, the chip is installed directly in the circuit board. Generally a socket has already been provided to make the operation easy, but occasionally some soldering may be required. Now you can give your 1541 disk drive "1571 speed."

Rather than give you more exagerated claims about how many times faster our ROM chip is compared to the slower cartridge versions, a comparison chart is supplied listing MEASURED loading times.

Program	Star-Dos	Reg. Load	Mach 5	Fast Load	MAGNUM LOAD
Pitstop II	?	144 sec.	43 sec.	41 sec.	**31 sec.**
Music Shop	?	105 sec.	105 sec.*	105*	**21 sec.**
Hitchhiker's Guide to the Galaxy	?	70 sec.	70 sec.*	N.G.**	**68 sec.**
On-field Football	?	149 sec.	66 sec.	63 sec.	**56 sec.**
EASY FINANCE I	?	58 sec.	13 sec.	13 sec.	**11 sec.**

$24.95 For 1541 or MSD Version

$39.95 for Combo Version

*Will not fast load - defaulted back to regular load

**Failed to load at all

D-CODER

- Translates any machine language program into easy-to-ready English descriptions with complete explanations of each command.
- Makes complete notations of all important memory locations accessed by the program (SID, VIC, MOS, KERNAL, etc.)
- Gives you three ways of accessing programs:
  1. **Will read and list programs from DISK**
  2. **Will read and list programs from MEMORY**
  3. **Direct user input (from magazines, etc.)**
- Can be used to locate and examine any machine language program's protection routines!
- Can be used to easily break apart machine language programs for study and examination!
- Printer option for complete hard copy listings!

You no longer need to be an EGGHEAD to read Machine Language.

$19.95

N-CODER

THE PERFECT COMPANION PROGRAM TO D-CODER!

Allows you to easily make changes in machine language programs . . . right on the disk!

- Rewrite ability allows code to be altered and then rewritten directly to the disk!
- Features sector-by-sector scrolling assembly language display of machine language programs!
- Notation of ASCII text equivalents for easy spotting of embedded text strings!
- Handy reference display of all assembly language commands and their ML numerical equivalents!
- Byte splitter for easy splitting of decimal addresses into low byte-high byte format!

$19.95

SOFTWARE PROTECTION HANDBOOK Third Edition! Now Available!

If you're **tired** of being harassed by **protected software** and too many **copy** programs, then this is the book for you! This 224 page manual covers the gamut from legalities to protection methods to step-by-step back up procedures. Now you can learn both how to **protect and unprotect** software! The techniques covered include copying cartridges to tape or disk, tape protection, and disk protection. Disk protection covers error numbers 20, 21, 22, 23, 27 and 29 **plus** single track formatting, header modification, header swapping, half track reading and writing, reading and modified bit densities, formatting illegal track/sectors, sync writing and **more!** The Third edition **explains**, tells how to **detect** and how to **write** them with included software.

C-64 Book Only .... **$19.95** US

Book & Disk of all Programs .... **$29.96** US

This manual does not condone piracy

*Shipping: $2.00

'TRACK TRAP' The 1541 Disk Expander!

• Supplement to the Software Protection Handbook •

The most unusual and innovative protection analysis tool for the Commodore yet! — **Not For Beginners** — This system expands your 1541 drive giving capability otherwise only possible for **professional** disk duplication equipment. Now you can create or analyze exotic forms of disk protection. **'D.O.S. Kings' Take Note!'** — Entire tracks of data can be read and written without regard to 'standard' sync and format. You are not longer limited to sector by sector searches. Whole track readouts reveal hidden data even when all or most of the sectors have been erased. Uncovers and writes data under errors, pulse coded sync or data, hidden data and access codes, multiple track densities and more! This supplemental manual covers the complete implementation of the track trap system including necessary software and hardware documentation.

Track Trap disk expanded manual .... **$9.95** US

Top Secret Stuff I and Top Secret Stuff II

Programed by Jim Drew

Are both collections of 20 programs per diskette (that works out to about $1.00 per program!) that help you explore and enhance your Commodore 64 and/or 128 and 1541 disk drive. Now you can unlock many secrets formerly known only to top machine language programmers by using these sophisticated "tools." If you have ever been curious about the inner workings of your computer system, now is your chance to dig in and find answers with the help of these programs. These collections of programs have gotten rave reviews from actual users, and we are sure that you, too, will be pleased.

The programs include on each diskette are listed below

TOP SECRET STUFF I

The Dock (view/repair disk contents)
Sync Checker (diskette)
Imbedded Track Number Creator
Disk Manipulation System
3 Minute Copy (backup program)
Diskette Matcher (compare sectors)
Unscratch A File (recover file)
View BAM (block allocation map)
1541 Read/Write Test
½ Track Reader
Header Reader (display disk header)
Sync Maker
Device Number Change (disk drive)
Electronic Arts Backup
Drive Mon (disk drive m/l monitor)
Diskette File Log (start-end address)
Write-Protect Sensor Test
Repair A Track (recover data)
Fast Format (10 seconds)
½ Track Formatter

$19.95

TOP SECRET STUFF II

RAM Test (test Computer RAM)
Copy $AOOO-$FFFF (under ROMS)
Display G.C.R. (All sector data)
Un-Write Protect (diskette)
Unnew Program
Wedge - $8000
Smooth Scroll (messages up screen)
Koala Dump (koala pad screen dump)
Disk Manipulation System
Disk Eraser (20 second clean wipe)
Split Screen (TWO screen colors)
Disk Protection System (stops copies)
Write Protect (diskette)
Boot Maker (autobook BASIC programs)
Wedge - $COOO
Diskmatcher II (high speed version)
No Drive Rattle (on reading errors)
3 Times Disk Drive Head Speed
Monitor Test (check video monitor)

$19.95

WAR GAMES AUTODIALER



1. **Auto Dial** will automatically dial a set of numbers you choose.
2. **Review Numbers** will review numbers that were answered by a computer.
3. **Save Numbers** will save numbers where a computer answered.
4. **Hardcopy of Numbers** will print out list of numbers where a computer answered.
5. **LOAD Numbers** will load in numbers to continue where it left off.
6. **Continue** will pick up dialing where it was interrupted.

$29.95

BBS

- 300/1200 Baud
- Remote access for sysop
- 2 levels of security for up and down load
- 7 rooms (read and write) with 4 security levels
- Secret highest level
- Open chalk board
- Auto message cycling
- Printer option



Plus the only BBS with all three (3) proticals X modem, new printer and midwest term, so anyone can up/down load.

Don't be fooled by cheap imitations. This is the most comprehensive system available anywhere. Now at a new low price.

$39.95

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. Canada orders must be in U.S. Dollars. VISA — MASTER CARD — C.O.D.

Programs for C-64 — $2.00 S & H on all orders

Software Submissions Invited

VISA / MasterCard

MegaSoft Limited

P.O. Box 1080, Battle Ground, Washington 98604

Phone 800-541-1541 • BBS 206-687-5205. After Hours Computer to Computer Ordering.

Reader Service No. 171

HOW TO CREATE WINDOWS

On big computers with window-oriented operating systems, the windows are used to show what's going on in several different programs running at the same time. Each program has its own video output, which it PRINTs into its own screen memory. Then the operating system copies that screen memory into that program's window in the main video memory.

It is possible on the 64 to do much the same thing. You could reserve 4K of memory, each K a separate pseudo-screen display. Then an interrupt routine would constantly copy the information from each pseudo-screen memory into a particular portion of regular screen memory at 1024.

When you POKE location 648 with the page number of one of the pseudo-screen memories, BASIC writes to that page. The interrupt program automatically copies what it finds there and puts it into a particular area of real screen memory at 1024. When BASIC is writing to screen 1, the CLEAR character will clear the whole 1K pseudo-screen memory; but in real screen memory, only screen 1 window will be cleared. BASIC never knows the difference.

But this uses up an awful lot of 6510 processing time, just copying one portion of memory into another portion, over and over. There are simpler ways.

For instance, as long as your windows are nothing but horizontal strips across the screen, with no vertical window boundaries, you could separate the windows by using scanline interrupts. You'll remember from last month that a scanline interrupt is a machine language program that executes whenever the TV raster scan reaches a particular line of the screen display. During that interrupt, your routine tells the computer to start finding screen memory in a different location.

You can do that as many times as you like on the screen. Furthermore, you can also make other changes—the background can be different, for instance, or the border can change colors—and those changes will also stay in effect until the next scanline interrupt.

That's what I did with the *Gypsy Pilot* program in the December *Ahoy!* Most of the screen showed screen memory located at 32K, which contained a field of stars (periods and asterisks). But the last two lines of the screen showed screen memory located at 35K, which held the strings showing fuel and supply levels.

(What I was really after, however, was a side effect: Since the sprite shape pointers are the last eight bytes of the 1K block that contains screen memory, when that scanline interrupt changed screen memory, it also allowed me to have the sprite shape pointers point to different shapes. In other words, below that scan line, my sprites changed shape. It happened that the shape block that those pointers pointed to consisted of nothing but zeroes. The result? The sprites all disappeared from that scanline on down!)

There is, however, another way to have many of the benefits of windows without all the machine language programming. And that is by using the Extended Background Color Mode built into the VIC-II video chip.

EXTENDED BACKGROUND COLOR MODE

In regular text mode, the character set consists of 2K of ROM. The first 512 bytes contain the shapes of all the regular capital letters, the numbers, and the punctuation marks. The second 512 bytes contain all the graphics characters. The third 512 bytes contain exactly the same letters, numbers, and punctuation marks as the first block—only reversed, with the letter taking the background color and the background taking the foreground color. The fourth 512-byte block is identical to the second, with its graphics characters, except that it is also reversed.

In other words, all the letters, numbers, and punctuation marks you use to communicate with the player during an all-text game are contained in the first quarter of character memory.

That means that with only 64 different screen character codes, screen memory could point to every single one of the letters, numbers, and punctuation marks you need. Yet there are four times that many possible screen codes—256 of them.

Ordinarily, screen codes (*not* ASCII codes) 0 to 63 point to the alphanumeric characters; screen codes 64 to 127 point to the graphics characters; screen codes 128 to 191 point to the reversed alphanumeric characters; and screen codes 192 to 255 point to the reversed graphics characters.

In bitwise terms, that means that the two high bits of each screen code decide which 512-byte block of character memory that shape will be drawn from.

In Extended Background Color Mode, however, *all* the character shapes are taken from the first 512-byte block—from the alphanumeric characters. An *A*, a reversed *A*, a shifted *A*, and a reversed-and-shifted *A* all have exactly the same shape: *A*.

Now that those two high bits don't have anything to do with choosing which block of character memory will be used, the VIC-II chip uses those high bits for something else: choosing a different background color.

A regular, unshifted, unreversed *A* will have the normal background color wherever you PRINT it on the screen. Its background color is selected by the number (from 0 to 15) at location 53281. (This is background color 0, and the *A* in screen memory has its two high bits set to 00.)

A shifted *A*, however, gets its background color from the number in location 53282. (This is background color 1, and the *A* in screen memory has its two high bits set to 01, for a value of 64 added to the screen code for *A*, which is 1.)

A reversed but unshifted *A* gets its background color from the number in location 53283. (This is background color 2, and the *A* in screen memory has its two high bits set to 10, for a value of 128 added to the screen code for *A*.)

And a reversed and shifted *A* gets its background col-

Continued on page 95

COMMODORE 64 COMPUTER

(Order Now)

$139.95

- **C128 Disks 79¢ ea.***
- **Paperback Writer 64 $34.95**
- **Commodore Graphics Printer $99.95**
- **13" Zenith Color Monitor $139.95**

CALL BEFORE YOU ORDER

COMMODORE 64 SYSTEM SALE

Deal 1	Deal 2
Commodore 64	Commodore 64
Com. 1541 Disk Drive	Com. 1541 Disk Drive
Com. 803 Printer	13" Zenith Color Monitor
$407	**$457**

PLUS FREE $49.95 Oil Barons Adventure Program

C128 COMMODORE COMPUTER

NEW

(Order Now)

$289.00

Plus FREE $69.95 Timeworks Wordprocessor.

- **340K 1571 Disk Drive $259.00**
- **Voice Synthesizer $39.95**
- **12" Amber Monitor $59.95**

PRICES MAY BE LOWER

COMMODORE 64 COMPUTER $139.95

You pay only $139.95 when you order the powerful 84K COMMODORE 64 COMPUTER! LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your computer that allows you to SAVE OVER $250 off software sale prices!! With only $100 of savings applied, your net computer cost is $39.95!!

* C128 DOUBLE SIDED DISKS 79¢ EA.

Get these 5¼" Double Sided Floppy Disks specially designed for the Commodore 128 Computer (1571 Disk Drive). 100% Certified. ***Lifetime Warranty.*** Automatic Lint Cleaning Liner included. 1 Box of 10 - $9.90 (99¢ ea.), 5 Boxes of 10 - $44.50 (89¢ ea.), 10 Boxes of 10 - $79.00 (79¢ ea.).

13" ZENITH COLOR MONITOR $139.95

You pay only $139.95 when you order this 13" ZENITH COLOR MONITOR. LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your monitor that allows you to save over $250 off software sale prices!! With only $100 of savings applied, your net color monitor cost is only $39.95. (16 Colors).

80 COLUMN COMMODORE GRAPHICS PRINTER $99.95

You pay only $99.95 when you order the 803 Commodore Graphics Printer, 60 CPS, Dot Matrix, Bi-Directional, Prints 8½" full size paper. Plug in direct interface included! LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your printer that allows you to SAVE OVER $250 off software sale prices!! With only $100 of savings applied your net printer cost is ZERO!

4 SLOT EXPANDER & 80 COLUMN BOARD $59.95

Now you program 80 COLUMNS on the screen at one time! Converts your Commodore 64 to 80 COLUMNS when you plug in the 80 COLUMN EXPANSION BOARD!! PLUS 4 slot expander! ***Limited Quantities***

80 COLUMNS IN COLOR PAPERBACK WRITER 64 WORD PROCESSOR $39.95

This PAPERBACK WRITER 64 WORD PROCESSOR is the finest available for the COMMODORE 64 computer! The ULTIMATE FOR PROFESSIONAL Word Processing, DISPLAYS 40 or 80 COLUMNS IN COLOR or black and white! Simple to operate, powerful text editing, complete cursor and insert/delete key controls line and paragraph insertion, automatic deletion, centering, margin settings and output to all printers! List $99.00. **SALE $39.95.** Coupon $29.95.

SPECIAL SOFTWARE COUPON

We pack a SPECIAL SOFTWARE DISCOUNT COUPON with every COMMODORE 64 COMPUTER, DISK DRIVE, PRINTER, or MONITOR we sell! This coupon allows you to SAVE OVER $250 OFF SALE PRICES!!

(Examples)

PROFESSIONAL SOFTWARE COMMODORE 64

Name	List	Sale	Coupon
Paperback Writer 64	$99.00	$39.95	$29.95
Paperback Database 64	$69.00	$34.95	$24.95
Paperback Dictionary	$24.95	$14.95	$10.00
The Print Shop	$44.95	$27.95	$26.95
Halley's Project	$39.95	$25.95	$24.95
Practicalc (spread sheet)	$59.95	$19.95	$14.95
Programmers Reference Guide	$24.95	$16.95	$12.50
Nine Princes in Amber	$32.95	$24.95	$21.95
Super Bowl Sunday	$30.00	$19.95	$17.95
Flip & File Disk Filer	$24.95	$14.95	$12.95
Deluxe Tape Cassete (plus FREE game)	$89.00	$44.95	$34.95
Pro Joystick	$19.95	$12.95	$10.00
Computer Care Kit	$44.95	$29.95	$24.95
Dust Cover	$ 8.95	$ 6.95	$ 4.60
Injured Engine	$39.95	$27.95	$24.95
Pitstop II (Epyx)	$39.95	$22.95	$19.95
Music Calc	$59.95	$14.95	$12.95
File Writer (by Codewriter)	$39.95	$29.95	$24.95

(See over 100 coupon items in our catalog)

Write or call for Sample SPECIAL SOFTWARE COUPON!

ATTENTION Computer Clubs

We Offer Big Volume Discounts

CALL TODAY!

PROTECTO WARRANTY

All Protecto's products carry a minimum 90 day warranty. If anything fails within 90 days from the date of purchase simply send your product to us via United Parcel Service prepaid. We will IMMEDIATELY send you a replacement at no charge via United Parcel Service prepaid. This warranty proves once again that ***We Love Our Customers.***

C128 COMMODORE COMPUTER $289.00

We expect a limited supply for Christmas. We will ship on a first order basis. This all-new revolutionary 128K computer uses all Commodore 64 software and accessories plus all CPM programs formatted for the disk drive. ***Plus FREE $69.95 Timeworks Wordprocessor.*** List $349.00. **SALE $289.00.**

340K 1571 COMMODORE DISK DRIVE $259.00

Double Sided, Single Disk Drive for C-128 allows you to use C-128 mode plus CPM mode. 17 times faster than 1541, plus runs all 1541 formats. List $349.00. **Sale $259.00.**

SUPER AUTO DIAL MODEM $29.95

Easy to use. Just plug into your Commodore 64 computer and you're ready to transmit and receive messages. Easier to use than dialing your telephone, just push one key on your computer! Includes exclusive easy to use program for up and down loading to printer and disk drives. ***Best in U.S.A.*** List $99.00. **SALE $29.95.** *Coupon $24.95.*

VOICE SYNTHESIZER $39.95

For Commodore-64 computers. Just plug it in and you can program words and sentences, adjust volume and pitch, make talking adventure games, sound action games and customized talkies!! FOR ONLY $19.95 you can add TEXT TO SPEECH, just type a word and hear your computer talk — ADD SOUND TO "ZORK", SCOTT ADAMS AND OTHER ADVENTURE GAMES!! (Disk or tape.) List $89.00. **SALE $39.95**

12" AMBER MONITOR $59.95

Your choice of green or amber screen monitor, top quality, 80 columns x 24 lines, easy to read, anti-glare! PLUS $9.95 for connecting cable. Com-64 or VIC-20. List $159.00. **SALE $59.95.**

PRINTER/TYPEWRITER COMBINATION $249.95

"JUKI" Superb letter quality, daisy wheel printer/typewriter combination. Two machines in one — just a flick of the switch. 12" extra large carriage, typewriter keyboard, automatic margin control and relocate key, drop in cassette ribbon! (90 day warranty) centronics parallel or RS232 serial port built in (Specify). List $349.00. **SALE $249.95.** **(Ltd. Qty.)**

13" RGB & COMPOSITE COLOR MONITOR $259.95

Must be used to get 80 columns in color with 80 column computers (C128 - IBM - Apple). (Add $14.50 shipping) List $399.00. **SALE $259.95.**

• LOWEST PRICES • 15 DAY FREE TRIAL
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL

PHONE ORDERS
8 a.m. - 8 p.m. Weekdays
9 a.m. - 12 noon Saturdays

• 90 DAY FREE REPLACEMENT WARRANTY
• OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail!
VISA — MASTER CARD — C.O.D. — No C.O.D. to Canada, APO-FPO

PROTECTO

We Love Our Customers

Box 550, Barrington, Illinois 60010

312/382-5244 to order

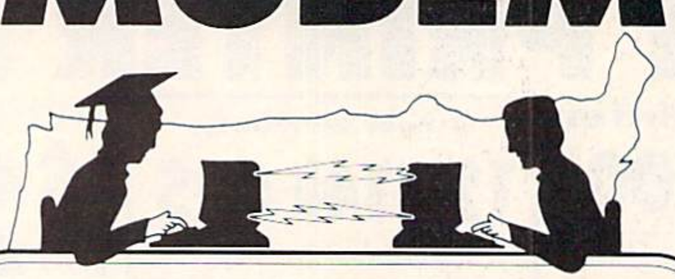
Commodore 64

MODEM

Commodore 64

Sale

SAVE



List $99.00

Sale $29.95

Coupon $24.95

Telecommunications

Lowest Price In the U.S.A.

List $99.00

Sale $29.95

Coupon $24.95

FOR CHILDREN·ADULTS·BUSINESS

Complete Auto Dial Telecommunications Package

"The only telecommunications package you will ever need."

(Exclusive Easy To Use Features)

FREE VIEWTRON DATABASE MEMBERSHIP

• Only Good Color Graphic Database Service in the U.S.A. (C-64)

Viewtron Software Plus First Hour FREE (See the Protecto Software Catalog On-Line) $9.95 value

• 300 Baud Modem • Auto Dial • Auto Answer • Upload & Download

Reach Out and Access Someone

- Educational courses
- Financial Information
- Banking at Home
- Popular Games
- News Updates and Information
- Electronic Shopping
- Research and Reference Materials

The Complete Telecommunications Package offers you all this plus ...

- Auto Log-on
- Dialing from Keyboard
- On-line Clock
- Capture and Display High Resolution Characters
- Download Text, Program or Data Files
- Stores on Disk Downloaded Files
- Reads Files from Disk and Uploads Text or Program Files
- Select Any Protocol (access almost any computer or modem)
- Plus Much, Much More

List $99.00 **Sale $29.95** Coupon $24.95

We are so sure this is the only telecommunications package you will need we will give you 15 days Free Trial.

Viewtron Membership sold separately — $9.95.

Add $3.00 for shipping handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail!

VISA MASTER CARD C.O.D.

No C.O.D. to Canada APO-FPO

Reader Service No. 176

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

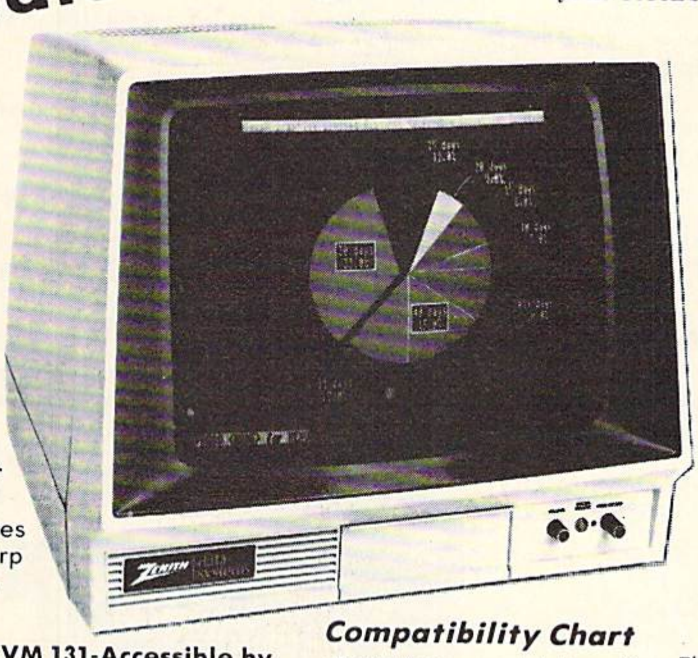
13" Zenith Complete Color Monitor Sale

• Composite • RGB • Sound

Home Computers VCRs Modular TV Tuners Video Games

Sale $139.95

LIMITED QUANTITIES



Excellent Color Reproduction & Special "Green Screen Only" Option

True color reproduction is achieved by a Zenith designed state-of-the-art integrated circuit chip that processes the composite video signal. A custom Zenith analog RGB direct drive gain control integrated circuit allows user-preference for the adjustment of picture drive and black level. Zenith's unique "Green Screen Only" feature eliminates all other colors so that monochromatic text material may be easily displayed in green on the black face screen.

Constant Intensity Character Definition Quality

Quality circuitry design generates crisp lines, pure colors, and sharp character definition for easy-to-read displays. DC-coupling permits the video display to retain its color balance from a single dot to a full screen of data. Even when room lighting changes, a "special light sensor" automatically adjusts the display brightness.

List $499.00

Sale $139.95

ZVM 131-Accessible by Many Popular Systems

The ZVM 131 is designed to interface with most personal computers, VCRs, video discs, video games and modular TV tuners that have either composite video or RGB direct drive outputs.

Compatibility Chart

Computer	Interfaces Via
Apple II	Composite
Aplus 3000	RGB
Apple III	RGB
IBM PC	RGB
Commodore 128	RGB/Composite
Commodore 64	Composite
Commodore Vic-20	Composite
TI 99/4	Composite
Atari 800	Composite
Atari 1200	Composite
Atari 1400	Composite

The ZVM 131 Sound Of Quality

The output sound level is externally regulated by a user-adjustment volume control. Use the Zenith quality sound system to monitor the modern audio capabilities of the computer generation.

Easy-To-Reach Front Access Controls

ZVM 131's 13" diagonal display screen can exhibit impressive graphics and intensely clear copy. Easy-to-reach front access user controls (picture, black level, color level, tint, sharpness, audio volume, background noise control) make display adjustment simple and fast. An LED power on indicator notifies the user when the monitor is operable.

Multiple Monitors On A Single Computer

The composite video "loop-thru" feature permits a single composite video source to drive several monitors at the same time. This allows easy display possibilities for multiple viewers in business and educational applications. No more crowding around a single terminal. Everyone enjoys a clear, unobstructed view of important data.

Connection Cables **RGB Cable — $19.95** C128, Aplus 3000 (Specify) **Composite — $9.95** Commodore, Aplus 3000, Atari (Specify)

This Is The Best Value Of The Century

Add $17.50 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $35.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 176

COMMODORE 64 139^{95}
COMPUTER

COMMODORE 128 289^{00}
COMPUTER

PLUS FREE $69.95 Word Processor





COMMODORE 64 SYSTEM SALE SAVE

LIMITED QUANTITIES

Deal 1	Deal 2
Commodore 64	Commodore 64
Com. 1541 Disk Drive	1541 Disk Drive
Com. 803 Printer	13" Zenith Color Monitor
$407	**$457**
PLUS FREE $49.95 Oil Barrons Adventure	PLUS FREE $49.95 Oil Barrons Adventure

Commodore C128 Computer $289.00 *

C128 1571 Disk Drive $259.00 *

SUPER OFFER **C128 COMPUTER** *SUPER OFFER*

FREE WORD PROCESSOR COUPON

(Expires 4-1-86) CM46

To Introduce you to the C128 computer we are offering you the finest word processor made, Word Writer II with Spell Checker by Timeworks. When you apply the $69.95 value word processor to your purchase price of the C128 at $289.00 **your net cost is only $219.05.** (1 Coupon per family)

C128 Word Writer with 85,000 word Spell Checker — An 80 column professional word processing system that includes a spelling checker and a built in calculator. Easy to use because of the full screen format, you can view the document on your screen as it will appear when printed. Pull-down menus mean that the user doesn't have to memorize commands. You press a key to activate a Word Writer feature and the program guides you through its proper use with logical and easy-to-follow prompts. The program has been designed to interface with Timework's Data Manager 2, a database program and Swiftcalc, a spreadsheet. Contains all the features you'll need for everyday word processing, plus more sophisticated features such as document chaining, form letter printout, page separations, horizontal and vertical scrolling and much more. (Disk) List $69.95.

* **Commodore C128 Computer $289.00.** This all-new revolutionary 128K computer uses Commodore 64 computer software, CPM Software, plus new advanced C-128 software. You pay only $289 for the C128 computer! **Less** the value of the **Special Software Discount Coupon** (see page 14 of our 64 page catalog) we pack with your computer that allows you to **Save Over $250** off software sale prices!! With only $100 of savings applied your net computer cost is $189.00. ***PLUS FREE $69.95 Word Processor.*** **

* **340K 1571 Commodore Disk Drive $259.00**. Double Sided, Single Disk Drive for C-128 allows you to use C-128 mode plus CPM mode. 17 times faster than the 1541, plus runs Commodore 64 software. You pay only $259.00 for the 340K 1571 Commodore Disk Drive. **Less** the value of the **Special Software Discount Coupon** (see page 14 of our 64 page catalog) we pack with your Disk Drive that allows you to **Save Over $250** off software sale prices! With only $100 of savings applied your net Disk Drive cost is only $159.00.

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 176

MUST LIQUIDATE At Far Below Dealer Cost!
TOTAL Personal Computer System

Factory Reconditioned with Factory Warranty!



Carries easily as a suitcase! Plugs into 115V outlet!

GREAT GIFT IDEA!

Sorry, we're not permitted to PRINT the famous brand-name. BUT, we CAN "tell all" if you call us TOLL FREE: 1-800-328-0609!

THE COMPUTER

Snap-on computer keyboard! 64K RAM, 20K ROM. Full-size typewriter keyboard. Upper and lower case letters, numerals, symbols, reverse characters. 2 cursor control keys, 4 function keys, programmable to 8. Music synthesizer with 3 independent voices, each with 9 octave range. Input/output ports accommodate . . . user, serial, ROM cartridge, joysticks, external monitor, phone modem.

Built-in disk drive! Intelligent high speed unit with 5¼" floppy disk recorder. 170K formatted data storage; 35 tracks. 16K ROM. Uses single sided, single density disk. Serial interface. Second serial port to chain second drive or printer.

Built-in color monitor! Displays 40 columns x 25 lines of text on 5" screen. High resolution. 320 x 200 pixels. 16 background, character colors.

Built-in ROM cartridge port! Insert ROM program cartridge. Multitude of subjects available in stores across the nation!

Original List Price **$995.00**

Liquidation Priced At Only **$388**

Item H-916-63631-00 Ship, handling: $20.00

THE PRINTER

Print method: Bi-directional impact dot matrix.

Character matrix: 6 x 7 dot matrix.

Characters: Upper and lower case letters, numerals and symbols. All PET graphic characters.

Graphics: 7 vertical dots — maximum 480 columns. Dot addressable.

Character codes: CBM ASCII code.

Print speed: 60 characters per second.

Maximum columns: 80 columns.

Character spacing: 10 characters per inch.

Line feed spacing: 6 lines per inch in character mode or 8 lines per inch selectable. 9 lines per inch in graphics mode.

Line feed speed: 5 lines per second in character mode. 7.5 lines per second in graphics mode.

Paper feed: Friction feed.

Paper width: 4.5" to 8.5" width.

Multiple copies: Original plus maximum of two copies.

Dimensions: 13"W x 8"D x 3¼"H. Wt.: 6½ lbs. **Power:** 120V AC, 60 Hz.

Original List Price: $200.00

Liquidation Priced At **$119**

Item H-916-63831-00 Ship, handling: $7.00

THE SOFTWARE

"Easy Script" One of the most powerful word processors at any price! Cut re-typing, create documents from standard paragraphs, do personalized letters, see and change a document before it is printed. Instruction manual has extensive training section that simplifies use . . . even for someone who has never used a computer or word processor before!

"The Manager" A sophisticated database manager for business or home use. Business uses: accounts payable/receivable, inventory, appointments, task manager. Home uses: mailing lists, home inventory, recipes, collection organizer, investment tracking, checkbook balancing. School uses: research article index, gradebook.

Mfr. Sug. Retail: $73.98

Liquidation Price **$24**

Item H-916-64011-03 Ship, handling: $3.00

BUY INDIVIDUAL UNITS OR GET THIS ULTRA-FAMOUS SYSTEM AT ONE LOW PACKAGE PRICE!

TOTAL Personal Computer System available at FAR BELOW dealer cost!

Original List Price **$1,293.00**

TOTAL PACKAGE PRICE **$488**

Item H-916-64011-02 Ship, handling: $24.00

Compatible with above Computer System (Not included in package price.)

JOYSTICKS (Set of 2)

Mfr. List: $59.90 pr.

Liquidation Price . . . . **$19 pr.**

Item H-916-63622-01 S/H: $4.00 pr.

64K MODEM (Factory New/First Quality) Limited 90-Day Factory Warranty.

Mfr. List: $124.95

Liquidation Price **$19**

Item H-916-63646-00 S/H: $4.00

VISA MasterCard

Credit card members can order by phone, 24 hours a day, 7 days a week.

Toll-Free: 1-800-328-0609

TeleCheck — Your check is welcome! No delays when you pay by check!

C.O.M.B. Direct Marketing Corp.
Authorized Liquidator
14605 28th Ave. N. • Mpls., MN 55441-3397

SEND ME THE ITEMS I HAVE LISTED BELOW

Sales outside continental U.S. are subject to special conditions. Please call or write to inquire.

Item No.	#	Item	Price	S/H
		TOTAL		

C.O.M.B. Direct Marketing Corp. Item H-916
14605 28th Ave. N./Minneapolis, MN 55441-3397

Send the items indicated at left. (Minnesota residents add 6% sales tax. Please allow 3-4 weeks delivery. Sorry, no C.O.D.)

☐ My check or money order is enclosed. (No delays in processing orders paid by check, thanks to TeleCheck.)

Charge: ☐ MasterCard® ☐ VISA®

Acct. No.__________ Exp.___/___

PLEASE PRINT CLEARLY

Name __________

Address __________

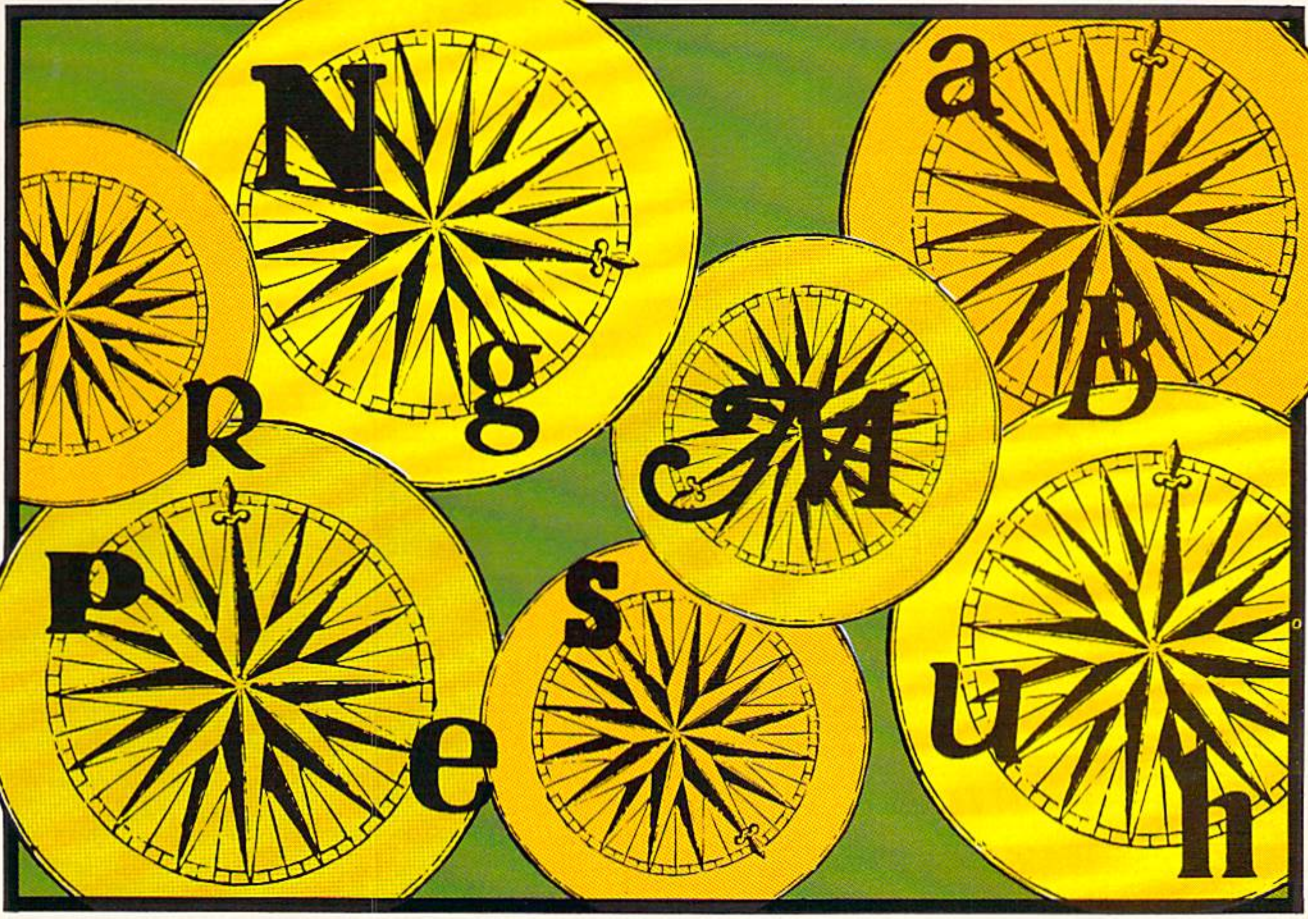
City __________

State __________ ZIP ______

Phone ( )

Sign Here __________

Reader Service No. 162



TREASURE WHEEL

For the C-64

By Bob Blackmer

Treasure Wheel is an entertaining two-player game loosely based on a popular television game show. Players take turns spinning the wheel for point values, then guessing letters, while trying to solve the puzzle. The puzzle can be from five different categories: person, place, thing, phrase, or title.

All information pertinent to the game is displayed on the game screen. At the top of the screen is the puzzle which uses dashes to show the blank puzzle. Near the lower left the category for the current puzzle is displayed. At the lower right is the total points each player has accumulated in the current game. At the upper right is the used letter board that shows all letters that have been selected in the game.

When your turn begins you press f1 to spin the wneel or f3 to solve the puzzle. If you select f1 the wheel will spin and come to rest on a point value (if you're lucky!). You then select a letter. If the letter is in the puzzle you receive the point value each time it appears in the puzzle and you take another turn, with the option of spinning or solving. If the letter you select is not the puzzle, play goes to the other player. If your spin results in LOSE A TURN, the turn also goes to the other player. If your spin lands on TILT you lose more than just your turn: all points scored in the current game are wiped out.

If you opt to solve the puzzle by pressing f3, the cursor will appear just below the puzzle. You must type the puzzle exactly, including spaces, to be correct. If you are correct you get 100 points plus your point total and the other player scores zero.

Running totals of games won and total points is seen after each game. The program keeps track of games in ten game sets. You also have the option of quitting after each game. The winner of the last game played goes first in the following game.

Be sure to turn the volume up, as sound is used for the spin of the wheel, when you TILT or LOSE A TURN, and for correct or incorrect responses.

After playing for awhile, your familiarity with the puzzles will justify changing them. Replace data statements with your own puzzles under each category. But remember to have a total of 20 puzzles per category, so use the REM statements to assist in organizing them.

SEE PROGRAM LISTING ON PAGE 107

Trusted Software and

Language Software

For Commodore Computers

ADA TRAINING COURSE



Teaches you the language of the future. Comprehensive subset of language. Includes: editor; syntax checker, compiler; assembler; disassembler, handbook. $39.95

ASSEMBLER/MONITOR



Macro assembler and extended monitor. Supports all standard functions plus floating point constants. Monitor supports bank switching, quick trace, single step, more. $39.95

BASIC COMPILER



Compiles complete BASIC language into either fast 6510 machine language and/or compact speedcode. Supports overlays and many extended BASIC commands. C-64 $39.95
C-128 $59.95

SUPER PASCAL



Full Pascal supports graphics, sprites, file management, more. Supports pointers, dynamic memory management, machine language. Compiles to fast 6510 machine code. C-64 $59.95
C-128 $59.95

FORTH LANGUAGE



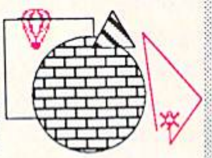
Based on Forth 79 (+ parts of '83). Supports hires graphics and sound synthesizer. Full screen editor, programming tools, assembler, samples, handbook. $39.95

MASTER



Professional development package for serious applications. Indexed file system, full screen & printer management, programmer's aid, multi-precision math, royalty-free runtime, more. $39.95

VIDEO BASIC



Add 50+ graphic, sound and utility commands to your programs with this super development package. Free distribution of RUNTIME version - no royalties! $39.95

XREF-128 & XREF-64

BASIC cross-reference



Indispensible tool for BASIC programmers. Finds all references to variables, constants & line numbers. Sorts in alphabetical order. C-64 $17.95
C-128 $17.95

Reference Books



ANATOMY OF C-64 Insider's guide to the '64 internals. Graphics, sound, I/O, kernal, memory maps, and much more. Complete commented ROM listings. 300pp $19.95

ANATOMY OF 1541 DRIVE Best handbook on this drive, explains all. Filled with many examples programs and utilities. Fully commented 1541 ROM listings. 500+pp $19.95 REVISED & EXPANDED

MACHINE LANGUAGE FOR C-64 Learn 6510 code & write fast programs. Many samples and listings for complete assembler, monitor and simulator. 200pp $14.95

GRAPHICS BOOK FOR C-64 Best reference, covers basic and advanced graphics. Sprites, Hires, Multicolor, 3D-graphics, IRQ, CAD, projections, curves, more. 350pp $19.95

TRICKS & TIPS FOR C-64 Collection of easy-to-use techniques: advanced graphics, improved data input, enhanced BASIC, CP/M, data handling and more. 275pp $19.95

1541 REPAIR & MAINTENANCE Handbook on the drive's hardware. Includes schematics & techniques to keep 1541 running. Align drive w/ & w/o scope. Large handbook size. $19.95

ADVANCED MACHINE LANGUAGE Subjects not covered elsewhere: video controller, interrupts, timers, I/O, extensions to BASIC. Tips for the serious programmer. 210pp $14.95

PRINTER BOOK C-64/VIC-20 Understand Commodore, Epson compatible printers & 1520 plotter. Utilities, screen dump, 3D-plot, commented MPS-801 ROM listings. 330pp $19.95

SCIENCE/ENGINEERING ON C-64 In-depth introduction to computers in science. Some topics covered are chemistry, physics, astronomy, electronics & others. 350pp $19.95

CASSETTE BOOK C-64/VIC-20 Make your cassette run faster than a disk drive! Cassette data-base, disk to tape backup, tape to disk, FastTape operating system. 225pp $14.95

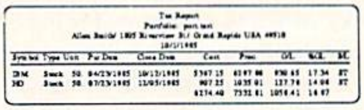
Productivity Tools

TECHNICAL ANALYSIS SYSTEM



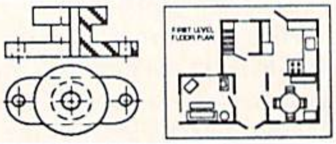
A sophisticated charting and technical analysis system for serious investors. By charting and analyzing the past history of a stock, TAS can help pinpoint trends & patterns and predict a stock's future. TAS lets you enter trading data from the keyboard or directly from online financial services. $59.95

PERSONAL PORTFOLIO MANAGER



Complete portfolio management system for the individual or professional investor. Allows investors to easily manage their portfolios, obtain up-to-the minute quotes & news, and perform selected analysis. $39.95

CADPAK



A deluxe graphics design and drawing package. Use with or without an optional lightpen to create highly-detailed designs. With dimensioning, scaling, text, rotation, object libraries, hardcopy and more. $39.95

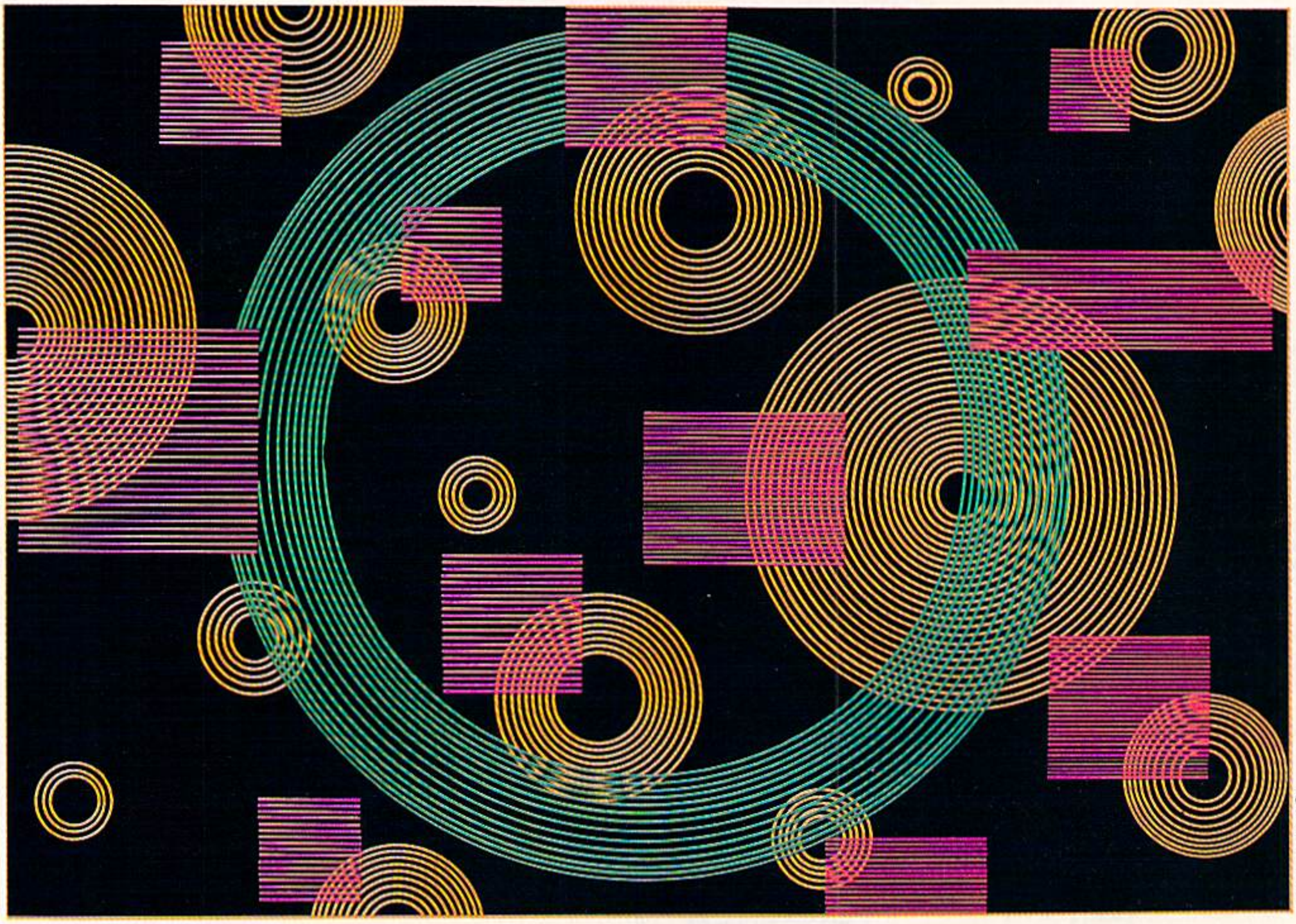
DATAMAT



Powerful, easy-to-use data management package using menu selections. Free-form design, 50 fields/record, 2000 records/disk. Sort on multiple fields in any combination. Complete selection and formatting for printing reports. $39.95

RUPERT REPORT

ELMAC



JAVIER ROMERO

The Computer Teaches Itself

BY DALE RUPERT

After a few thousand quick games with itself, ELMAC had become a fairly competent NIM player. ELMAC is certainly not the world's fastest learner. Perhaps after studying the structure of ELMAC you will be able to modify its cerebral organization to improve its playing habits and its learning ability.

ELMAC is an acronym for Elementary Learning Machine. The program of which ELMAC is comprised plays a simple game called NIM. ELMAC keeps track of each of its moves throughout the game. At the end of the game, if ELMAC won, each move is rewarded, and if ELMAC lost, the last move ELMAC made is punished.

Each move that ELMAC makes is chosen randomly. Some moves are more likely to be randomly selected than others. The probability of a particular move being selected depends upon the current board layout and the number of times in which that move has been rewarded or punished in the past. After playing enough games, ELMAC's probability of randomly choosing a good move is very high and its probability of picking a poor move is very low or zero.

At first this concept of learning by modifying probabilities may seem almost trivial. Further thought may convince you that perhaps this is indeed somewhat analogous to the way that you learn something new.

C-64 and C-128* OWNERS:



Now YOU too can get the "Software Edge!"

Do you know the full contents of all your Disks in your Software Library?

Stop playing "Hide-and-Seek" with your Disk software! Let your Commodore 64™ "memory" do all the work for you. Easy Manual and Program Menu-Screens make everyone a "pro" in 1 day!

Here's a small "PEEK" at SOME of the FULLY-INTEGRATED POWER of PRODUCTIVITY at your fingertips. . .

MASTERDISK©:

- a SORTED Master Catalog of ALL your Disks (even those un-listable "protected" Directories: Names, ID's, File info, etc.)
- Search + List any File/Disk/ID in seconds, automatically, in any of SEVEN WAYS.
- Transfer ML-Programs or Files from Disk to Disk (CATEGORIZE your software)
- Stop disaster before it strikes: change duplicate ID's fast, without total re-copy.
- Restore accidentally-scratched files in a flash, with VERIFICATION that they're OK.
- Works with 1 or 2 Single Disk Drives (1541, MSD, etc.) with or without Printer (Commodore or ASCII).
- Handy-Reference Card for HEX/ASCII/ BINARY-TO-DECIMAL VALUE TRANSLATION + FORMULAS, included FREE.
- All this + MUCH MORE, for just $34.95. (U.S.A.), $45.95 (Canada)

NEW!! - MASTERDUAL©,
"Son of MASTERDISK©" . . .
same power, but with
Dual Drive Speed!

- Same Catalog Features as MASTERDISK©
- Same Disk repair functions, too
- For all C-64 & C-128* systems with Dual Disk Drives (4040, MSD, etc.)
- Utilizes all Dual Drive functions for maximum speed and ease of use
- FREE Handy Reference Card for HEX/ASCII/ Binary to Decimal Translation + Formulas
- Experience FULLY-INTEGRATED POWER for just $29.95 (U.S.A.), $39.95 (Canada)

***MASTERDISK© and MASTERDUAL©** are also compatible with the NEW C-128 (in C-64 Mode)!

Don't MESS with anything LESS! ***OUR CUSTOMERS WRITE:***

"Been meaning to write and let you know how much I've enjoyed it . . . So, let me say that MASTERDISK is EXCELLENT." — Middletown, NY

". . . just the thing I need." — APO, NY (Germany)

"Hope to get my [MASTERDISK] yesterday." — Maple Grove, MN

"I am impressed with your product and would like to be able to endorse it for TPUG." — Ontario, Canada

". . . nothing else even comes close!" — Burlington, IA

We ship within 24 hours! Send Money Order or Check to:

INTEGRATED-SOFTWARE SYSTEMS
P.O. Box 1801, Ames, IA 50010
VISA/M.C./C.O.D., CALL **(515) 233-2992**
9 am - 9 pm (Central Time) Mon.-Fri.
Add $3 Shipping/plus $2 Foreign, C.O.D.
IA Residents add 4% Sales Tax.

IN CANADA: Cricket Distribution Co.,
P.O. Box 953, Cobourg, Ontario, Canada K9A 4W4
VISA/M.C./C.O.D., CALL **(416) 372-3692**
9 am - 6 pm (Eastern Time) Mon.-Fri.
Add $3 Shipping/plus $2 C.O.D.
Ontario Residents add 7% Sales Tax

Whether you have tens or hundreds of disks, now YOU can be the MASTER of your TOTAL Software Library!

Reader Service No. 168

HEURISTIC OR ALGORITHMIC?

It should be clear from the outset that ELMAC is not programmed to be a good NIM player. It is possible to write a program which is a perfect NIM player. That is not the intent of this article or of the accompanying program.

ELMAC is a heuristic NIM player as opposed to an algorithmic NIM player. The heuristic player is able to adapt and to improve. In essence it learns from its mistakes. The algorithmic player uses a prescribed formula (or algorithm) for determining its moves. The two concepts are certainly not mutually exclusive. A playing algorithm for example might include some heuristic rules, and the heuristic player's moves are generally rewarded or punished according to some algorithm.

The game of NIM involves two players who take turns removing sticks (or other objects) from the playing board. Initially there are three rows containing three, four, and five sticks respectively. Each player in turn takes one or more sticks from any one row. The last player to take a stick is the winner. A game might look like this:

First player:	0	4	5
Second player:	0	4	1
First player:	0	1	1
Second player:	0	1	0
First player:	0	0	0

Initially the board configuration is 3 4 5. The first player removed three sticks from the first row, leaving the board with none in the first row, four in the second row, and five in the third row (0 4 5). Each board display shows the number of sticks in each row *after* the named player has made a move.

The second player took four sticks from the third row leaving 0 4 1. The first player then removed three sticks from the second row. The second player took the remaining stick from the third row. The first player then removed the last stick on the board, thereby winning the game.

There are many variations on this game, including differing numbers of sticks and rows and even whether the last stick drawn belongs to the winner or the loser. Martin Gardner discusses the game of NIM and algorithms for winning it in *The Scientific American Book of Mathematical Puzzles & Diversions* (Simon & Schuster, 1959). Martin Gardner also describes elementary learning machines in *The Unexpected Hanging and Other Mathematical Diversions* (Simon and Schuster, 1969). The machines which Mr. Gardner describes were built by readers of *Scientific American* in the early 1960s from match boxes and colored beads. Both books are high recommended reading.

HERE'S ELMAC!

The program which implements ELMAC may be used in several different ways. (Refer to ELMAC Plays NIM on page 105). The easiest way to begin is to select the mode in which the computer plays against itself. Simply press C at the opening screen. In this mode, ELMAC randomly selects its moves from lists of moves which have been weighted by rewards and punishments. ELMAC's opponent simply picks a move at random, with each legal move equally likely to be selected.

The statistics displayed at the top of the screen show ELMAC's win/loss record, as well as its current and longest winning streaks. After a few hundred games, ELMAC usually has a winning average significantly greater than 50%.

Once ELMAC has become smart enough, you might enjoy playing against it. Press any key to interrupt the current mode of play, and the opening menu appears. Press P when you are ready to play against ELMAC. The first player is chosen at random. If ELMAC went first, the screen displays the current board layout after its move, otherwise the initial rows of three, four, and five are shown. You are prompted to select the row and the number of items to be removed from that row. Only legal moves are accepted.

At the end of the game, the winner is announced. You may press any key and ELMAC's statistics will be updated before proceeding into the next game. If you want to allow ELMAC to pick up some more intelligence on its own, simply press M instead of a row number or an item number. You will be returned to the menu where you may select C.

ELMAC's intelligence consists of an array of valid moves for every possible board configuration called the "move array." You may save the move array to disk at any time during the game by pressing a key to stop the game and then selecting S. You must specify a filename. It must be different from any other filename on the disk.

You may also load a move array from disk at any time. Essentially this provides a brain transplant for ELMAC. Press any key to interrupt the game then select L. You must specify the filename of a move array which was previously saved with the S command.

ELMAC creates a move array initially with all legal moves having equal probabilities. You may look at the move array at any time by pressing any key to stop the game and then selecting D. Each board configuration and the associated move probabilities are scrolled. We will discuss the format of this array shortly. Pressing any key during the scrolling causes it to stop. You are asked if you want to save this array, just as if the S key had been selected.

Do not be discouraged by the length of the program. There are many comments throughout the program. Any REM statement which is preceded by a colon need not be typed (although REMarks will be very useful if you make any modifications to this program!). The program is written in a reasonably modular form so that you may easily understand and change it.

ELMAC'S STRUCTURE

The data structure to implement ELMAC is somewhat complex. Perhaps the easiest way to understand it is to look at some specific examples.

The move array is dimensioned with DIM A(3,4,5,12). The first three subscripts correspond to the numbers of

...Announcing

THE
COMMODORE USERS

Ahoy! Access Club!

FREE TO ALL SUBSCRIBERS





An incredible offer from the publishers of *Ahoy!*...The ***Ahoy!* Access Club.** The ***Ahoy!* Access Club** offers members: discounts, rebates, coupons on products and services for your Commodore, a Commodore calendar of events, and much, much more! Members will also receive "The Clipper," an exclusive bi-monthly news bulletin. But the best thing about the ***Ahoy!* Access Club** is that it's **FREE!** to all *Ahoy!* subscribers. Subscribe today! and along with your first issue of *Ahoy!* you'll receive your official ***Ahoy!* Access Club** membership card...your ticket to super savings and bonuses you can only get from *Ahoy!* Don't miss out on this fantastic offer. The ***Ahoy!* Access Club**...if you own a Commodore — this club is for you!

THE COMMODORE MAGAZINE THAT OFFERS YOU MORE

sticks in each row for every board layout. The fourth subscript represents the probability of each move being made from that board configuration.

Consider the initial values of A(1,3,2,1) through A(1,3,2,12):

1 0 0 – 1 1 1 0 – 1 1 0 0 0

(The –'s are placed between A(1,3,2,3) and A(1,3,2,4) as well as between A(1,3,2,7) and A(1,3,2,8) to help with this discussion.) When the board has one stick in the first row, three sticks in the second row, and two sticks in the third row (1,3,2), ELMAC chooses its next move from the values listed above.

The first three elements of the array correspond to removing one, two, and three sticks respectively from the first row. The next four elements of the array correspond to the probability of taking one, two, three, and four sticks from the second row. Similarly, the probabilities of removing one through five sticks from the third row are represented by the last five elements of the move array.

The first element, A(1,3,2,1), has a value of one, indicating that it is possible to remove one stick from the first row. Notice that the second and third elements, A(1,3,2,2) and A(1,3,2,3), are zero. This means that the probability of taking two or three sticks from the first row is zero. That's to be expected since there is currently only one stick in the first row.

The next group of four elements (1 1 1 0) shows that it is possible to remove either one, two, or three sticks, but not four, from the second row. The final group (1 1 0 0 0) indicates that there is an equal probability of removing either one or two sticks from the third row, and no probability of taking three, four, or five sticks. Again this is reasonable, since these array elements correspond to a board with only two sticks left in the third row.

One other element of the move array tallies the total number of move probabilities for each particular board layout. In the example above, that element is A(1,3,2,0). Its value is initially six, which is the sum of the twelve probabilities listed above for this board configuration.

Let's assume the board configuration is 1 3 2, and that it is ELMAC's move. ELMAC looks at the total move count in A(1,3,2,0) which is six (refer to line 560 of the program). ELMAC selects a random number between one and six, say four. Lines 620 through 660 determine what type of move that is.

Recall that the twelve elements of A(1,3,2,P) for P equal 1 through 12 are

1 0 0 – 1 1 1 0 – 1 1 0 0 0

ELMAC picked four, and the fourth value of one in the list is the third element in the second group, A(1,3,2,6). Consequently ELMAC removes three sticks from the second row. (If you are not totally confused, you are a genius!)

If ELMAC eventually wins this game, the value of A(1,3,2,6) will be incremented by one. Likewise A(1,3,2,0) will be incremented. Now A(1,3,2,P) for P equal to 0 through 12 has values:

7 – 1 0 0 – 1 1 2 0 – 1 1 0 0 0

The next time that ELMAC makes a move from this board configuration, it will pick a random number from one to seven. It then steps through the twelve moves, tallying probabilities until it reaches a count of seven. It should be clear that if ELMAC selects a random value of four or five, it will make the same move it made last time. In other words, ELMAC is twice as likely to choose move A(1,3,2,6) as it is to choose any of the other possible moves. That is exactly how ELMAC learns through rewards.

To see how bad moves are punished, consider the board configuration 2 0 0, that is, only two sticks remain in the first row. If this is the first time ELMAC has seen this particular board, the move array A(2,0,0,P) for P equal to 0 through 12 appears as

2 – 1 1 0 – 0 0 0 0 – 0 0 0 0 0

ELMAC can remove either one or two sticks from the first row, and both moves are equally likely.

Obviously taking both sticks is the only intelligent move, but ELMAC isn't very bright yet. If ELMAC takes only

one stick from the row, the other player removes the remaining stick, and ELMAC loses the game. In that case, such a foolish last move by ELMAC is punished by reducing its probability to zero. That means that A(2,0,0,1) is now set to zero.

The next time ELMAC sees the board 2 0 0, its only possible move is A(2,0,0,2). It takes both sticks from the first row. ELMAC wins the game and A(2,0,0,2) is incremented. With the program as it is currently written, only ELMAC's final move of a losing game is punished. Once a move's probability has been reduced to zero, that move is never made again.

The array M( ) is updated with every move ELMAC makes. Its structure is fairly straightforward. M(MN,1), M(MN,2), and M(MN,3) represent rows one, two, and three at the time of ELMAC's MNth move. M(MN,0) stores the type of move (from 1 to 12) that ELMAC made.

For example, assume the board had two sticks in the first and third rows and three sticks in the second row when ELMAC made its second move. M(2,1), M(2,2), and M(2,3) have values 2,3,2 respectively, representing the board configuration. If ELMAC removed two sticks from the third row, M(2,0) equals 9. (Recall that move values 1 to 3 correspond to taking 1 through 3 sticks from the first row, values 4 through 7 represent the removal of 1 through 4 sticks from the second row, and values 8 through 12 represent taking 1 through 5 sticks from the third row, respectively.)

At the end of each game, if ELMAC won, M( ) is used in lines 1150 through 1200 to reward each move. The last move made by ELMAC is always kept in the variable MV, and the last board configuration is stored in X0, Y0, and Z0 for rows one, two, and three in lines 710. Lines 1040 through 1060 punish ELMAC's last move if it lost.

TODAY NIM, TOMORROW THE WORLD

After you have learned to play NIM, see if the way you learned is not somewhat analogous to the way ELMAC learns. Typically you will first remember and avoid moves near the end of the game which are fatal, just as ELMAC does.

There are many possibilities for your own exploration into this concept of learning machines. I suggest saving ELMAC's move arrays after various numbers of games. I saved the move array after one thousand and ten thousand games, calling them ELMAC.1K and ELMAC.10K respectively. It is interesting to study the move array (use the D command) to see how bad moves are eliminated and probabilities of good moves are enhanced.

The D command scrolls all elements of the move array. The first three numbers displayed in reverse video are the board configuration. The first number in the next line is the total count of move probabilities. The next twelve numbers are the individual move probabilities. One interesting characteristic to look for in the move array is the opening move A(3,4,5,−). A perfect algorithmic machine will remove two sticks from the first row to guarantee a win. Consequently A(3,4,5,2) should eventually become the only allowable opening move for ELMAC to make. The resulting board configuration is 1 4 5.

You might investigate other methods of reward and punishment. Perhaps later moves in a winning game should be rewarded more (or less) than earlier moves. Perhaps more than just the last move should be punished.

It is possible to find a far more efficient data structure than I have used here. Martin Gardner mentioned that one of his readers built a NIM-playing machine from only 18 match boxes (for three rows of three objects each). That learning machine supposedly played an excellent game of NIM after only 30 games. ELMAC's move array represents 120 match boxes, and it plays a lousy game even after 300 rounds.

It would be handy to have a graphical display of ELMAC's winning record. When ELMAC plays the random computer, it is really a waste of time to show both players' moves. A graph of number of games versus number of wins would probably be more informative.

ELMAC's data structure should be readily adaptable to other games such as tic-tac-toe or four-by-four checkers. The Commodore 64 does not have enough memory to store the move array for tic-tac-toe as implemented in ELMAC. Some clever programming, and taking advantage of the symmetries and rotations of the board, should make a tic-tac-toe program possible.

If ELMAC has no valid moves for a particular board configuration, it resigns. After 15,000 games, it had no possible moves when faced with two sticks in each of two rows. Indeed, a perfect player would beat ELMAC under these conditions, but since ELMAC doesn't necessarily play against perfect players, perhaps an element of "oh what the heck" should be added to ELMAC. Let ELMAC make a valid random move even if the move array says there are no winning moves available.

The situation is similar to the old joke mentioned by Martin Gardner in which two chess-playing computers were face to face. After the one computer moved its first piece, the second computer furiously thought for a while, and then simply resigned. Eventually it seems that ELMAC would have just one valid opening move, and if its opponent made that move first, ELMAC would call it quits.

There are many interesting, intriguing, and entertaining aspects to this idea of machines that teach themselves. Hopefully ELMAC will encourage you to explore. I would enjoy hearing about what you discover. □

SEE PROGRAM LISTING ON PAGE 105

KIDS AMERICA

Ahoy! is pleased to congratulate

Jens Palm (Croton-on-Hudson, NY)
Bhuvanesh Frininison (Buffalo, NY)
Annise Winger (Henrietta, NY)

winners of subscriptions to *Ahoy!* courtesy of the Kids America radio program. Check your local AM and FM listings for time and station.

THIS ISSUE—
TWO COMMODORE SOFTWARE SURVEYS!

THE DIGITAL BLUE YONDER

Flight Simulation on the C-64

By Bill Kunkel and Arnie Katz



No, we didn't print it upside down—it's Sierra's Stunt Flyer.
READER SERVICE NO. 143

"Flying by the seat of your pants" is an old aviation expression. It refers to situations in which circumstances force a pilot to navigate by instinct alone, without the aid of instruments. Today, that phrase could just as easily refer to the armchair aviators who regularly patrol the electronic skies in front of their Commodore home computers.

Flight simulators have come into their own over the past two years. Such games satisfy the current taste for strategy, while they provide the type of real-time excitement which once made action games so popular. Users who drop in at the Commodore Aerodrome can earn their wings in anything from a Cessna to a Strike Eagle. There are helicopters, biplanes, and even a space shuttle waiting for gamers with the will to take the throttle.

Early Flights

Flight simulation programs have been available for the Commodore 64 from the very first. The pioneering titles which are still on the market show their age, but may interest some ardent aviators.

IFR Flight Simulator (Academy) puts the user inside a Cessna 172RG. Unfortunately, the design strikes a poor balance between instruments and scenery. The principle display is almost entirely filled by gauges, the artificial horizon, and similar navigational devices, while the pilot must squint through a window no larger than the airspeed indicator to see what's happening outside. *IFR Flight Simulator* replicates the physics of flight, but not the experience of sitting in the cockpit of a small plane.

Solo Flight (MicroProse) and *Flight Path 737* (Artworx), two other early entries, are fun despite limited graphics. Both are simple take-off-and-land simulations which emphasize actually flying an aircraft. Good as far as they go, *Solo Flight* and *Flight Path 737* are primitive by current standards. However, the latter title comes on cassette, which makes it the only choice for Commodore owners who do not have disk drives.

If the boom in Commodore 64 flight simulators can be traced to one event, it's the publication of the C-64

PERSONAL BEST

A Guide to Individual Sports Games on the C-64

By Bill Kunkel and Arnie Katz



Summer Games *demands good timing.*
READER SERVICE NO. 144

Okay, sports fans, get out those wrist bands, clean up that old pair of track flats, and tie a rolled up bandana around your head. It's time for some one-on-one sports action. Whether played against another human video athlete or versus the computer, events like track and field, golf, and tennis offer some of the hottest competition in computer gaming.

The Olympics have become this best-known arena for individual athletics, thanks to satellite telecasts. Not surprisingly, the quadrennial event has inspired several computer programs.

Decathlon (Activision) was the first title which simulated track and field. It includes all of the 10 events which comprise the greatest challenge in the world of sport: 100m dash, long jump,



Flight Simulator II *features full onscreen instrumentation, variable weather conditions, realistic avionic systems, and choice of day, dusk, or night flying. READER SERVICE NO. 145*

PERSONAL BEST
A Guide to Individual Sports Games

400m, shot put, 110m hurdles, 1500m, javelin, discus, pole vault and high jump.

The player manipulates the joystick to direct the onscreen decathlete. Most of the events require the gamer to jiggle the stick rapidly to make the character on the screen move faster. You'll really need those wristbands with this disk.

Summer Games (Epyx) demands pinpoint timing rather than manic joysticking. The user must execute a precisely timed sequence of stick movements to put the onscreen athlete through his paces.

Summer Games simulates swimming, diving, gymnastics, and skeet shooting, as well as a trio of track and field contests. The electronic Olympian can practice single events or run the gauntlet of all eight. The disk retains a record of high scores which the player can access through the program's main menu.

THE DIGITAL BLUE YONDER
Flight Simulation on the C-64

edition of *Flight Simulator II* (SubLOGIC). Bruce Artwick's design, based on a program developed for the IBM PC, created a sensation when it made its debut, and it's still a pretty fantastic disk. *Flight Simulator II* is one of the all-time best-selling entertainment programs, and dominated the Billboard list for more than a year.

Flight Simulator II's biggest attraction is its astonishing completeness. It features full on-screen instrumentation, a choice of day/dusk/night flying, variable weather conditions, realistic avionic systems and a flying range which spans the entire continent.

The simulation is based on the Piper 181 Cherokee Archer, a small single-engine plane with non-retractable landing gear. The scenery disk packaged with the program generates the proper out-the-window visuals for landings and take offs at any of 80 airports in the New York, Chicago, Seattle and Los Angeles areas. SubLOGIC promises more scenery disks for those who want to visit other places.

As a bonus, *Flight Simulator II* includes a "World War I Ace" mode. This variation streamlines the play-mechanic of the primary program and applies it to biplane flight, circa 1917. The changes reduce the accuracy of the simulation, but provide a lot more excitement. Skillful flying is still crucial, but a would-be ace must also demonstrate deadeye marksmanship against the enemy.

The scenery disk adds familiar landmarks such as the Statue of Liberty to the out-the-cockpit-window view. The accurately placed landmarks are attractively rendered, and the overall graphics are surprisingly good for a program which must assign so much memory to simulation elements.

Flight Simulator II does a fine job of recreating the dynamics of piloting on the home computer screen, but apart from the "World War I Ace" mini-game, it is essentially a purist's approach to the subject. Following it onto computer store shelves are several titles which immerse the computerist in more intense scenarios than zipping through the friendly skies from New York's LaGuardia to Chicago's O'Hare.

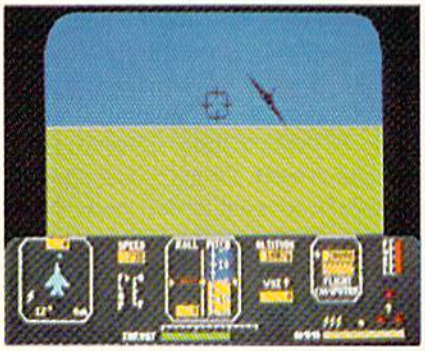
The skies are considerably less friendly in *F-15 Strike Eagle* (MicroProse). This solitaire program places the user in the cockpit of a modern jet fighter which must complete missions right out of today's scariest newspaper headlines.

Although *F-15 Strike Eagle* makes extensive use of the keyboard as an input device, the joystick controls the most important functions. The computer pilot can climb, descend, bank, drop bombs, and fire the machine gun without touching a key.

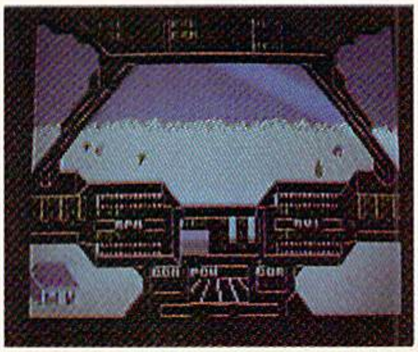
F-15 Strike Eagle casts the gamer as a combat flyer. The missions send this lightning-fast aircraft to places like the Persian Gulf and Haiphong Harbor. Invariably, these strongholds are protected by hostile MIGs which are more than willing to engage in combat.

The visuals are decent, if unexceptional. Several types of ground installations are presented, which gives some visual variety to the bombing runs.

F-15 Strike Eagle is more concerned with stimulation than simu-



Jet Combat Simulator: *four modes. READER SERVICE NO. 146*



Super Huey: *ideal for two players. READER SERVICE NO. 147*

This solid simulation comes in very fancy dress. The graphics are outstanding, and the animated torch-lighting is one of the most impressive openings in entertainment software.

The success of *Summer Games* prompted Epyx to release a sequel, *Summer Games II*. It features an array of eight somewhat more exotic events, including the difficult triple jump, bicycling, kayaking, fencing, and even equestrian jumping.

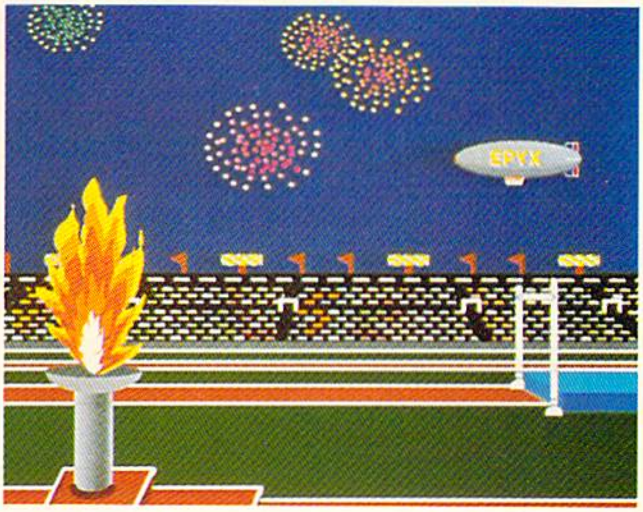
Each event has its own theme music, and the graphics are, if anything, better than in the original. Judges' scorecards appear on the screen at the conclusion of each round of competition to rate the performance.

Summer Games II can be linked to its predecessor to create a mammoth 16-event program. Its newly created closing ceremony is the perfect ending.

Fans of Olympic simulations might like to augment the Epyx events with *S-Games* (Michael Crick, 2845 140th Ave. NE, Belleview, WA 98005). In addition to the more conventional events—100m sprint, 110m hurdles, and diving—*S-Games* includes a couple of truly unique competitions which are not available on any other title: archery and weightlifting.

A new publisher has updated and re-released this contest, originally known as *HES-Games*. Crick led the original design team, 3-2-1 Software, which developed the program, and he has upgraded it intelligently and effectively.

While the track events use the joystick-jiggle play-mechanic, the approach to archery and pumping iron is more reminiscent of *Summer Games'* control system. In the archery



Summer Games II *can be linked with its predecessor to form one 16-event program.* II*'s events are more unusual ones, including kayaking, fencing, and the triple jump. READER SERVICE NO. 148*

---

lation. Its exciting combat scenarios more than compensate for any technical imperfections.

Fans of this type of air action will also enjoy *Jet Combat Simulator* (Epyx), written by Digital Integration. This program, which utilizes joystick and keyboard input, focuses on realistically duplicating flight procedures and problems.

Jet Combat Simulator, like *F-15 Strike Eagle*, uses the McDonnell Douglas F-15 as its main aircraft. The disk includes four separate modes: landing practice, flight training, air-to-air combat practice and actual combat. The user can vary the difficulty by pre-program factors such as blind landings, crosswinds-turbulence and the skill level of the computer-directed enemy.

Several programs which nominally fit into this category stress action. *Mig Alley Ace* (MicroProse) and *Hellcat Ace* (MicroProse) are short on aeronautical verisimilitude, but are as exciting to play as any arcade game.

Stunt Flyer (Sierra) represents a third type of airplane simulation, the "air circus." This disk, a creation of the French design group Nice Ideas, involves more than just preventing the craft from digging a hole in the runway. The accent is on performing fancy aerobatics.

Stunt Flyer puts the pilot inside the most maneuverable of all stunt planes, the Pitts Special. Although the game is quite detailed, it eschews factors like "torque," which would do little more than complicate the already demanding test. The aerodynamics of trick flying are the main event.

Stunt Flyer has three separate programs: airshow, training, and competition. The airshow provides an external view of a computer-guided plane which performs any of 20 stunts on command. Training allows the user to fly freestyle or attempt a specific stunt, then view an external-perspective replay. A digital wing-walker who feels comfortable enough at the controls can try the competition. Any high-flying Commodore owner who completes the compulsory sequence is eligible to enter Sierra's nationwide competition.

Other Flights

The airplane has always been the favorite subject for flight simulations, but some designers have gone in other directions. After all, there are other types of flying machines.

Paul Norman's *Super Huey* (Cosmi), for example, attempts to reproduce the experience of piloting a UH-IX, an ultra-modern surveillance and reconnaissance helicopter. The program concentrates heavily on piloting the chopper, although combat is a tangential element.

Super Huey is ideal for cooperative play. Helicopters are much more difficult to fly than airplanes, so there's plenty to do. The command control system uses both the joystick and the keyboard, which facilitates division between pilot and co-pilot.



Jet *is just up the stream. READER SERVICE NO. 149*



Winter Games *offers skiing, skating, and bobsledding events, including the unusual biathlon, which combines cross-country skiing with marksmanship. READER SERVICE NO. 150*

PERSONAL BEST
A Guide to Individual Sports Games

competition, the video Robin Hood must consider wind speed and direction in addition while aiming for the bullseye. Weightlifting is broken down into two-hand snatch and clean and jerk. The player lifts more poundage by maintaining smooth timing and good rhythm.

S-Games allows participants to save great performances to a preformatted disk for subsequent replay. The animation and graphics are good, if not quite state-of-the-art.

The most recent, and most visually spectacular, Olympic software product is *Winter Games* (Epyx). Players can now revel in simulations of skiing, skating, and bobsledding, enhanced with the best sound and graphics in the series so far. Action Graphics had a couple of tough acts to follow, but the design team earned straight A's for its work on this one.

Winter Games begins with a real flash event, the hot dog aerial. As an onscreen skier takes off into the wild blue yonder, the gamer executes a series of dazzling flips, kicks, and dives. The biathlon, one of the most unusual events, combines cross-country skiing with marksmanship. In speed skating, the user determines the stride of the onscreen athlete, who is paced by a computer-controlled rival. The ski jump simulation presents two views of the action. The main display offers a long-range view, while a special window shows a close-up.

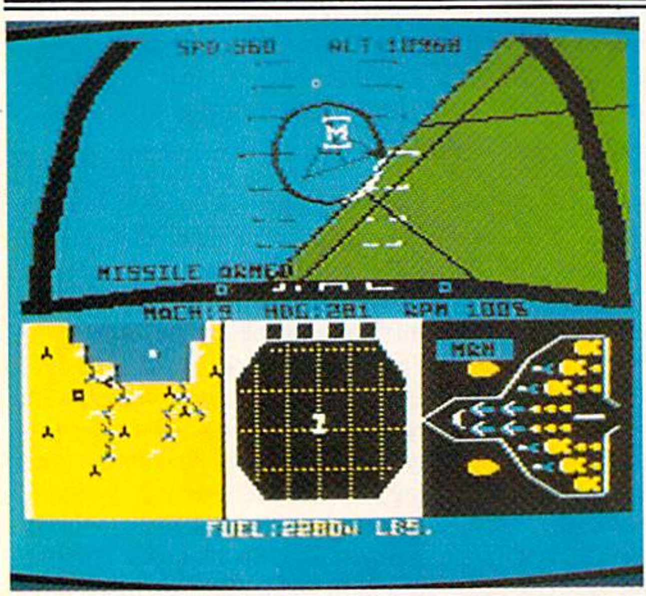
The most astonishing events are figure skating and the bobsled. The musical accompaniment, a perfect scene-setter for the onscreen Peggy Fleming, gives the Commodore sound chip a brisk workout. Among the movements possible in the skating competition are: camel spins, sit spins, double axels, triple axels, double lutzes, triple lutzes, and camels-into-sit spins.

The bobsled race employs a split display to give an overview of the action as well as a quasi-first person perspective of the run. (The bobsled appears in the foreground as a guiding device.) The sense of "being there" is intense as the sled whooshes around turns and beneath the overpasses to the cheers of the spectators.

Winter Games has an opening ceremony, but no similarly ornate close. That's a sure sign that Epyx will eventually delight computer sports fans with a *Winter Games II.*

Go for the Green

Golf is a sport which has always



F-15 Strike Eagle *is more concerned with stimulation than simulation, sending players on dangerous combat missions to MIG-defended enemy strongholds. READER SERVICE NO. 151*

THE DIGITAL BLUE YONDER
Flight Simulation on the C-64

The graphics in *Super Huey* are blocky, which makes the surveillance scenarios seem pointless. Watching isn't much fun when there's nothing worth looking at.

When you get tired of flying around this world, there's always outer space. Steve Kitchen's *Space Shuttle* (Activision), although somewhat dated, still has a lot to offer. Players manipulate three axis controls to take the shuttle into orbit, dock with a satellite, re-enter Earth atmosphere and land her on the appropriate runway.

The most eagerly awaited flight simulator on the artificial horizon is the follow-up to *Flight Simulator, Jet* (SubLOGIC), from the Bruce Artwick crew. *Jet* promises to be as revolutionary as its predecessor. The already published IBM edition includes take-offs from an air strip or a carrier, choice of weapons, and a selection of dogfights and strike missions in either an F-16 Fighting Falcon or F-18 Hornet.

The introduction of the Commodore 128 could be the prelude of a new generation of more exact flight simulators. And considering the steady improvement in the category so far, one might say the sky's the limit. □

WHITE HOUSE COMPUTER

P.O. BOX 4025, WILLIAMSPORT, PA 17701

In PA: 1-717-322-7700
1-800-351-3442

HOURS: Monday thru Friday 9:00 am till 6:00 pm

VISA and MC Accepted 4%

DISK DRIVES

C-64

1541 .... 185.95
1571 .... 249.95
1572 .... 375.95
MSD Single 229.00
MSD Dual 459.00
Indus GT 199.00
Cardco SCD-1 ... 199.00

INTERFACES

Tymac
Connection/2K ... 62.95
Cardco +G 59.95
Cardco B 37.95
OR–D
GPC w/2K 55.95
Grappler CD 89.95

PRINTER PAPER

Lazor 1000 shts ... 15.95
Lazor 2500 shts ... 29.95

DISKETTES ★

SKC

SS/SD 10.50
SS/DD 13.50
DS/DD 15.50

MAWELL

MD-1 .... 15.95
MD-2 .... 21.95

ELEPHANT

SS/SD 13.50
SS/DD 15.50
DS/DD 19.50

INNOVATIVE CONCEPTS

Flip-N-File
10 .... 3.50
15 .... 8.50
25 .... 18.50
w/lock
50 .... 17.50
50 .... 23.50
ROM Holder ... 17.50

KOALA

Pad Disk 39.95
Pad ROM 45.95

★ PRINTERS ★

C-64

MPS 801 .... 179.95
802 .... 208.95
803 .... 175.95
MCS 801 .... 314.95
DPS 1101 .... 314.95

CITIZEN

MSP-10 325
MSP-15 479
MSP-20 469
MSP-25 629

EPSON

RX80 219
LX80 249
FX80 369
JX80 529
RX100 369
FX100 550

C. ITOH

8510 A 285
8510 BC1 329
8510 BC2 389
8510 BP1 339
8510 SP 389
8510 SR 429
8510 SCP 459
8510 SCR 477
1550 BCD 485
1550 P 445
A10-20P 465
F10-40P/S 875
F10-55 P/S 1059
F10 Tractor 169
F10 Sheet Feeder 349

MANNSEMAN TALLY

Spirit 80 249
MT 160L 495
MT 180L 595

STAR MICRONICS

SG10 210
SG15 379
SD10 345
SD15 450
SR10 485
SR15 585
Powertype 309

CARDCO

LQ 1 365
★**LQ 3 .... 209**

PANASONIC

1090 187
1091 231
1092 389
1093 585
3151 455

LEGEND

880 209
1080 229
1380 269
1385 305

OKIDATA

182 .... 219.95
84 .... 640.95
92 .... 349.95
93 .... 560.95
Imagewriter 425.95

MONITORS

AMDEK

300G 119
300A 129
310A 145
Color 300 235
Color 500 345
Color 600 399
Color 700 469
Color 710 539

TEKNIKA

MJ-10 175
MJ-22 249

XTRON

Comcolor I 189

ZENITH

122 .... 74.00
123 .... 74.00

BMC

Amber 82.95
Green 79.95

COMMODORE

C-1702 .... 199.95
C-1802 .... 204.95
C-1902 RGB 259.95
C-1901 Monchrome 129.95

Dealers Inquiries Invited ★



COMPUTERS

C-64 ..... 149.95

C-128 ... 275.95

SOFTWARE ★

BATTERIES INCLUDED

Paper Clipw/Spell Pk ... 69.95
Paper Clip 59.00
Home Pack 32.95
Bus Card II 124.95
B180 Board 99.95

BRODERBUND

Print Shop 29.95
Graphics Library 18.95
B.S. Writer 32.95

CONTINENTAL

Home Accountant 27.95
Tax Advantage $27.95
GET BOTH FOR $49.95

VIDEO CASSETTE ★★ RECORDER $275.95

Symphonic VHS/Remote

- 105 Channel Cable Ready
- 14 Day / 1 Program Timer
- 12 Pre-set Channels
- Auto Rewind

Plus much, much more!

CHECK OUR PRICES!

MODEMS

C-1600 .... 44.95
C-1650 .... 54.95
C-1660 .... 54.95
C-1670 .... 185.95
300/1200 Band

Cardco
MOD-1 .... 59.95
C-64
MPP-1064 ... 65.95
Tele Learning 49.95

DUST COVERS

Mitey Mo ... 59.95
Westridge ... 69.95
Compuserve Starter Kit ... 21.95

WE CARRY A FULL LINE OF:
Batteries Inc., Broderbund, Contineal, Hess, Infocom, Innovative Concepts, Koala, LJK, Microbits, Micropose, Microtek, O.S.S., Professional S.W. Scarborough, S.S.I., Synapse, Time Works, Wico.

POLICY: No deposit on COD orders. FREE freight on all prepaid cash orders over $300 in the continental USA. APO and FPO add $5 per hundred. For priority Mail add $8 per hundred. PA Residents add 6% sales tax. Defective products must have Prior RA number. Schools net 15.

Reader Service No. 186

MAKER BILLBOARD MAKER BILLBOARD MAKER
MAKER BILLBOARD MAKER BILLBOARD MAKER
MAKER BILLBOARD MAKER BILLBOARD MAKER

Thanks for waiting ... BILLBOARD MAKER, the Ultimate Graphic Utility, IS FINALLY HERE!!!

IMAGINE TAKING YOUR USUAL 8½ inch by 6½ inch output AND ENLARGING IT TO 4 FEET BY 3 FEET !!!!

* * * SIMPLY INCREDIBLE * * *

Billboard Maker brings you the capability of enlarging your computer graphic masterpieces into attractive poster and "BILLBOARD" size reproductions.

"If you've ever wanted to use your own graphic pictures and never could, BILLBOARD MAKER opens a whole new world for you!!! You'll be able to take your original computer art and create effects you never thought could be done. BILLBOARD MAKER is a MUST addition for graphic applications never before available.

- *DUMP shaded standard and ENLARGED graphic pictures to your printer QUICKLY and EASILY*
- *Overlay specially designed TEXT fonts and ICONS on your enlargements - A TRUE SIGN MAKER*
- *Allows re-drawing of details to enlarged sections resulting in incredible resolutions!!!*
- *Easy to use utility to bring your pictures into your OWN PROGRAMS!!!*
- *CONVERT picture file formats to other product formats*
- *SMOOTH those ragged lines with the amazing "OPTIMIZER"*
- *FLIP pictures for iron-on applications*
- *STRIP color from your picture*
- *REVERSE picture to negative - CROP and FRAME pictures*
- *AND MANY MORE GREAT FEATURES PROFESSIONAL RESULTS - EASY and FUN to USE*

BILLBOARD MAKER works as a stand alone product as well as with picture files generated by the following products;

ANIMATION STATION	*COMPUTER EYES*	*FLEXIDRAW*	*PRINT SHOP*
BLAZING PADDLES	*DOODLE*	*KOALA*	*SUPERSKETCH*

EQUIPMENT REQUIRED: Commodore 64, TV or Monitor, 1541 Disk Drive, Printer (required for print dumps)

DISTRIBUTOR/DEALER INQUIRES CALL RELEVANT SOFTWARE MARKETING 914/683-5158

Reader Service No. 207

adapted well to computer. Each "play" is a distinct event, and the physics of the flight of the ball are easy to express as mathematical formulas.

Computer "hackers" will quickly get into the swing of *Professional Tour Golf* (Strategic Simulations), which incorporates elements of both strategy and stat replay. After selecting a bagful of clubs, the user guides such links legends as Slammin' Sammy Snead and Jack Nicklaus around either of two courses, complete with sand traps, trees, and water hazards.

A special option allows the computerist to generate a simulation of any famous golfer or even fashion a model based on your own golfing prowess. Strategic Simulations promises supplemental disks containing new courses and golfers.

On-Court Tennis (Gamestar) applies a similar form of artificial intelligence to net sports. Each gamer controls an electronic racqueteer who plays with a distinctive style. Some hug the baseline, others charge the net at every opportunity, and still others reply on a cannonball serve to score points.

The program automatically moves the electronic athletes around the court based on the styled selected prior to the opening serve. A joystick-activated control system lets the computerist call each shot, once the onscreen surrogate gets into the proper position. This makes *On-Court Tennis* more a test of strategy than hand-eye coordination. Proper timing is important, but it's definitely subordinate to selecting the right stroke for each situation.

Space limitations have forced this survey to neglect two major categories: auto racing and martial arts. Rest assured, both will receive their justice in upcoming issues of *Ahoy!* Meanwhile, these "personal best" programs should keep solitaire and head-to-head gamers at their computers for many enjoyable hours of individual competition. ☐

FEATURED NEXT MONTH:
Educational Games for the Commodore 64



A sheriff who can uphold the Law of the West *without bloodshed scores points and receives additional credit on the evaluation screen at the game's end. READER SERVICE NO. 152*

LAW OF THE WEST
Accolade
Commodore 64
Disk; $29.95

It takes more than a fast draw to keep peace in the town of Gold Gulch. A successful lawman must show insight, tact, and just the right degree of firmness to keep the streets and alleys safe.

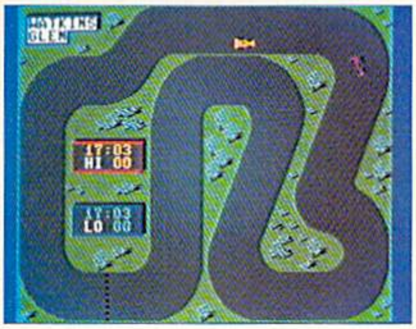
Gunfight programs predate the dawn of the home computer age. Every videogame system had at least one such cartridge. Stripped of minor differences, they all boil down to armed figures on opposite sides of the display shooting at each other. *Law of the West* breaks the mold by fitting the shoot-outs into an adventure game context.

Law of the West is played as a series of encounters, each of which involves a meeting on the street between the sheriff, controlled with the joystick by the gamer, and a computer-directed character.

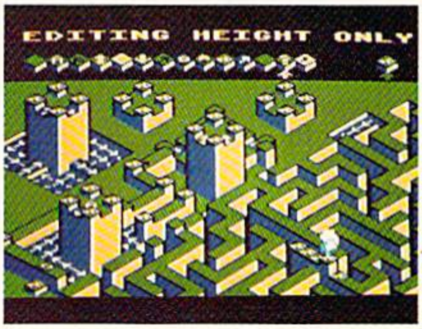
An encounter consists of two types of activity: role-playing and action. Each scene begins with the hero, a large figure in the foreground of the display, looking up the street at a beautifully drawn western scene. The view changes for each round of play, and may include a bank, the stage depot, a saloon, or the train station.

A computer-controlled character enters the scene to the strains of original theme music and, in most cases, saunters up to the sheriff. A line of text highlighted in red appears below the picture. This is the opening line of the conversation. The would-be law officer selects the desired response from an onscreen menu of four possibilities.

What happens next depends on the tone of the conversation. A hostile response may lead to more heated exchanges which culminate in a showdown. On the other hand, a good sheriff can talk seemingly violence-bent gunslingers into better behavior by clever choice of comments. A non-player character can become



On Track: *a long way from LeMans. READER SERVICE NO. 153*



Lode Runner's Rescue: *46 mazes. READER SERVICE NO. 154*

your buddy or your dueling partner, depending on how he or she is handled. And a newly met friend may warn you about the robbery which invariably becomes the next episode. Warnings frequently make the difference between dropping the bad guy and taking a slug yourself.

A sheriff who takes care of a situation without artillery scores points and receives additional credit on the evaluation screen at the end of the game. This recap indicates how well the gamer has performed during the day as a symbol of justice. Categories considered are: maintenance of authority, crooks captured, romantic exploits, bad guys sent to boot hill, recoveries from bullet wounds, innocent people killed, and the number of crimes committed.

Sometimes, talking isn't enough. Pushing the joystick forward draws the gun and engages the aiming cursor. The hero must try to keep order in Gold Gulch without blasting innocent civilians, and an itchy trigger finger gets static from townspeople and a low rating.

Alan Miller, who made quite a reputation as a videogame designer for Activision, returns to computer software for the first time since he created *Basketball* for Atari in 1979. Though he's lost none of his flair for innovation, this program's limitations suggest that Miller is still adjusting to this latest turn in his career. Despite the outstanding artwork, perhaps the best found in any Commodore 64 game, *Law of the West* lacks the depth of a good adventure. Most experienced computerists will exhaust its play-value within just a few games.

The frills are very impressive, but they don't make *Law of the West* more challenging. Some of the memory devoted to the undeniably superb audiovisual effects might have been spent more profitably on enlarging the scope of the menu-driven interaction. It's an inspired concept, but this program barely explores the possibilities.

It's a shame to knock a game with as many positive features as *Law of the West*, but its defects make it impossible to recommend wholeheartedly. Alan Miller took a good shot, but he's missed the bullseye on this one.

Accolade Software, 20863 Stevens Creek Blvd., #E, Cupertino, CA 95014 (phone: 408-446-5757).

—Arnie Katz

ON TRACK MODEL CAR RACING
Gamestar
Commodore 64
Disk; $24.95

Pure competition is the focal point of this hard-driving contest for one or two armchair Andrettis. As the title indicates, this is *model* car racing, not a replica of the 24 hours of Le Mans. So despite the use of famous courses, *On Track* conjures up more images of radio-controlled vehicles buzzing around a parking lot than of heavy tires squealing down the Indy brickyard.

As in many Gamestar products, a series of option screens which define the game precede actual play. After selecting one or two players, the contestants pick between manual and automatic shifting. Manual, which is recommended only for the experienced, requires drivers to change between high and low gears to coax maximum performance out of their automobiles.

Next, each player picks one of three car/driver tandems. Artificial intelligence makes Mario Sandduni a speed demon and gives A.J. Cactus' machine the best handling characteristics. A chart in the instruction folder makes it simple to understand the differences.

The following screen lets the gamer choose from among 10 tracks. These range from Road America, with its hairpin turns, to the lengthy straightaways of Fuji International. Each course can be either paved or dirt, with the latter producing more unpredictable handling.

The final screen establishes the speed ("pro" or "amateur") and the victory conditions. Drivers can compete for time in five- or 10-lap contests or they can try to go the furthest in 12 or 24 hours.

Each course occupies one display screen with no scrolling. Both cars are always in sight, which eliminates the need to stop and reset the cars when one builds too big a lead. On the other hand, the visuals are pretty tame compared to some other racing simulations. The accent here is definitely on head-to-head rivalry, not scenery.

A computerist steers the car with the joystick. Pushing the joystick to the left causes the car to execute a similar turn, while moving the handle to the right produces the exact opposite effect. Steering is done from the perspective of a driver riding in the onscreen vehicle, but this becomes second nature after a few laps.

A race car accelerates automatically, but the driver must press the action button to apply the brake. Maneuverability improves markedly when a car really gets rolling; however, noone can negotiate some of the crazy turns on these tracks without using the brake at least occasionally.

On-Track Computer Model Car Racing is no easier to master than other Gamestar titles, which means computerists won't wear it out in a hurry. Newcomers may at first feel like they're steering a bug on a block of ice, but liberal use of the "practice" mode proves that keeping all four wheels on the road consistently is possible.

Is *On Track* too simple? Yes and no. It is certainly not going to excite those who want a detailed simulation or flashy graphics. Yet it does offer a chance for continuous, head-to-head racing, and that's a rare and noteworthy feature.

Gamestar, Inc., 1302 State St., Santa Barbara, CA 93101 (phone: 805-963-3487). *—Steve Davidson*

THE FOURTH PROTOCOL
Bantam Software
Commodore 64
Disk; $34.95

Frederick Forsyth's best seller of the same name inspired this spy thriller for computerists. Thanks to its innovative play-mechanic and twisting, three-episode plotline, the adventure has the same urgency and excitement as the book. Gamers are swiftly drawn into a tale of international intrigue, espionage, and nuclear threat, as they become John Preston, British Intelligence Officer.

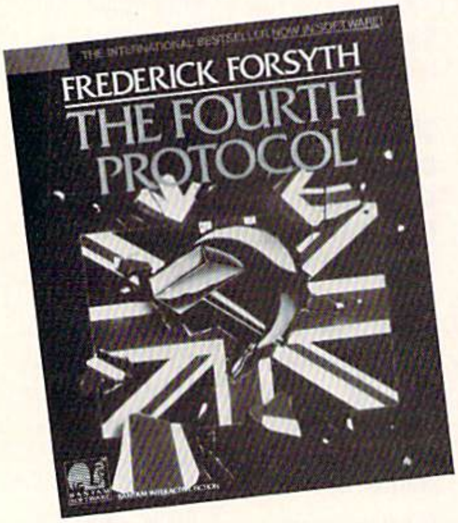
The Nuclear Non-Proliferation

Treaty signed by Britian, the U.S., and the Soviet Union in 1968 forbade these countries' dissemination of nuclear weapons technology to other nations. The Treaty contained four secret protocols. The fourth protocol forbids the signatories from ever smuggling nuclear weapons into another country for covert detonation.

Russian Central Committee intelligence officers have hatched up a scheme to weaken NATO by destabilizing the Western Alliance. To do this, they plan a nuclear disaster in England, to be blamed on the American military presence in the U.K. They figure this will result in England's withdrawal from NATO and, subsequently, perhaps even a Soviet takeover of Western Europe. The computerist, as John Preston, counterintelligence agent, stumbles upon this plot and must track down the saboteurs before they fire the nuclear device.

The game is divided into three separate episodes. In the first, Preston must learn who stole the NATO documents, why, and to whom they leaked the information. Time passes quickly as Preston checks reports, hunts through the files, assigns watchers to suspects, and performs his routine duties.

If Preston digs up the mole in British intelligence, he advances to the second part. In this phase, Preston has to locate the bomb which the Soviets have already smuggled into England. The short, final section features a free-for-all battle between Special Air Service troops and KGB agents, as Preston tries to disarm the bomb.



A twisting, three-episode plotline.
READER SERVICE NO. 150

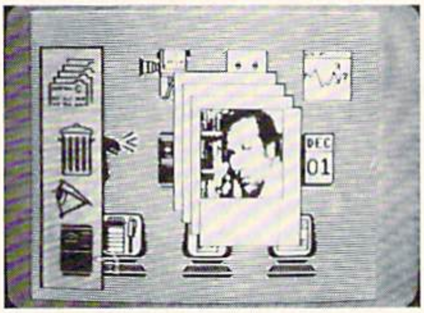
The first two game sections are icon-driven. Instead of typing orders, as in parser-driven adventures, the computerist activates the appropriate picture on the screen with the pointer. Game one, The Nato Documents, puts Preston at the well-equipped desk of his office. Using the keyboard to move the cursor around a ring of icons, the super-agent reads memos and reports, uses the telephone, contacts headquarters, and orders surveillance of suspicious characters.

A somewhat different, more action-oriented set of icons is utilized in part two, The Bomb. They allow Preston to pick up, use, or drop objects, examine items, and listen to informants. The onscreen compass allows Preston to move through the city to gather the facts he needs. Most of the travel around London is via the Underground, and some adventure gamers will feel that the designers of *The Fourth Protocol* have lavished entirely too much detail on the intricacies of rapid transit. A wise agent will hunt up the address of the places which he needs to visit and get to them by cab whenever possible.

The SAS Assault, the final section, is more of an extended climax than a full-blown episode. It is based on a small parser which lets the gamer enter brief commands in an effort to defuse the bomb and its associated safeguards before the device explodes. This segment is much like traditional illustrated adventures with a full-screen drawing of the room in which the bomb is actually hidden.

The finale is fun, but it's the icon-driven sections which make *The Fourth Protocol* a unique game. Putting the options right at Preston's fingertips ends the principle frustration of parser-oriented adventures: expressing what you want to do in words the machine understands. This is a fast-paced adventure, not a head-scratching word-hunt.

The Fourth Protocol is not especially difficult and is mercifully free of the convoluted logic puzzles which



The Fourth Protocol*'s fast-paced first two sections are icon-driven.*

characterize more primitive adventure programs. The emphasis is on the experience of becoming a secret agent on a life-or-death mission, and few titles use role-playing more effectively. The player must become John Preston, at least for the space of a few enjoyable hours, to thwart the plot and preserve world security. The well-prepared documentation, which thankfully includes an extensive glossary, acclimates the player to the setting and provides a few helpful clues.

The Fourth Protocol is more than an innovative design, it is a thoroughly enjoyable piece of adventure software.

Bantam Electronics, 666 Fifth Avenue, New York, NY 10103 (phone: 212-765-6500). *—Joyce Worley*

LODE RUNNER'S RESCUE

Synapse
Commodore 64
Disk; $29.95

It isn't always easy to be a devoted daughter. This is especially true if your dad is that legendary leaping secret agent known as the Lode Runner.

The hero of Doug Smith's award-winning *Lode Runner* (Broderbund) and *Championship Lode Runner* (Broderbund) is sitting in a Bungeling Empire jail. His daughter, the equally high-flying Alexandra, must jump, run, and dodge through 46 three-dimensional mazes to reach the cell containing Lode Runner and liberate him. As in the earlier episodes of this series, guards patrol the labyrinths. Their touch costs one of the four precious lives with which Alexandra starts her quest.

Female computer gamers have complained, and rightly, that there's a shortage of heroines in entertainment software. Alexandra is an im-

portant step toward closing the gender gap. Apart from a slight fear of heights, which prevents her from jumping more than two levels down under normal conditions, Alexandra is every bit as much a daredevil as her famous father. And she's got at least one trick up her sleeve which even her old man would envy.

The rules folder suggests that the player turn the joystick diagonally (clockwise) so stick directions correspond better to Alexandra's on-screen movements. She can walk on level ground, climb up one step, or jump down two without expiring. (The dauntless daughter can fall greater distances safely if she lands on an elevator, exit, or water square.)

Alexandra hops over the adjacent square when the gamer presses the action button while pointing the stick in the appropriate direction. She can leap over any space which is two steps high or less. Taller barriers cause the lady rescuer to bounce back to the square from which she began her leap.

Lode Runner's pit-digging ability caused command control problems in the Commodore editions of the first two titles in the series. It proved hard to implement the hero's ability to dig a trap for the guards either in front or behind with a conventional one-button joystick.

Lode Runner's Rescue won't make Commodore owners feel like second-class citizens. Designer Joshua Scholar has dropped this feature and, instead, endowed Alexandra with the ability to temporarily change size by eating one of the mushrooms scattered around the playfields.

A snack turns Alexandra into a giantess who can leap across two squares or fall three height levels without danger. A timer made of mushrooms in the upper right hand corner monitors the nine-second "tall" period. When the heroine shrinks to normal size, a tone sounds to alert the gamer.

Lode Runner always scrounged for gold, but his daughter collects the keys which are scattered around each maze. To advance to the next screen, Alexandra must scoop up all the keys and reach the exit square before the bonus clock runs down to zero. Picking up a key adds 50 points to the score and a like amount to the bonus. When the computerist completes a round, the game adds all points remaining on the bonus counter to the total score.

The three-dimensional perspective gives the format pioneered in *Lode Runner* a new lease on life. Few entertainment programs boast a prettier batch of playfields. And in most cases, they're as challenging as they are good-looking.

In the long run, the best thing about *Lode Runner's Rescue* is the screen editor mode. As in the original *Lode Runner*, the user can create customized playfields when the regulation ones have been conquered. The process is a little more difficult, since would-be designers must visualize in three dimensions, but no special knowledge of programming is required.

It might be reaching to say that *Lode Runner's Rescue* surpasses the first *Lode Runner*, which is still a classic. Yet it is definitely a worthy successor and an entertaining gaming experience in its own right.

Synapse, 17 Paul Drive, San Rafael, CA 94903-2101 (phone: 415-479-1170). *—Arnie Katz*

DECISION IN THE DESERT
MicroProse
Commodore 64
Disk; $39.95

The North African campaign of World War II has fascinated military historians and wargamers alike for more than 40 years. The sweeping movement of small armies over vast stretches of territory, punctuated by intense periods of fighting, makes this theater stand out in a war characterized by a heavy concentration of forces clashing over restricted terrain.

The "accelerated real-time" play-routine is well-suited to the subject. *Decision in the Desert* does not proceed in alternating turns like most computer wargames. Commanders in this one- or two-player simulation issue a steady stream of orders, which units execute as game time passes. Speed is not essential, since a divi-

sion can do only so much during each four-hour period, but the time pressure certainly builds excitement. Pressing the "F" key circumvents this feature by pausing the game. This is necessary if there are two human generals, but solitaire players should probably go easy on the "strategy breaks."

Decision in the Desert contains five basic scenarios. Included are Sidi Barrani (the early British victories against Italy), Operation Crusader (the Axis seige of Tobruk), Gazala (Rommel's victory at Tobruk), First Battle of Alamein (Rommel's campaign after Tobruk), and Alam Halfa (Montgomery's defense of the Nile)

Most of the scenarios have "what if?" variants which allow computerists to study these battles from new perspectives. Operation Crusader, for example, has an option which lets the German general assault Tobruk before the British arrive. Another intriguing variant gives military buffs a chance to see if Rommel could have beaten Field Marshal Montgomery had he received more supplies to back his bold stroke.

Solitaire gamers will appreciate the excellent digital general incorporated into *Decision in the Desert.* It plays a strong game for either side and can be adjusted to give novices and veterans alike a competitive tussle.

Limited intelligence is available as an option in both solitaire and head-to-head games. A commander can only see enemy units which are within line-of-sight of his or her own troops. It introduces a realistic degree of suspense and uncertainty and helps give this program the feel of the actual campaign.

Although authors Sid Meier and Ed Bever have included the kind of sound and graphics which greatly enhance the appeal of *Decision in the Desert* to those who don't ordinarily gravitate toward wargaming, this is a very detailed simulation. The basic strength of each unit varies during the course of play, depending on the weather, defensive terrain, the experience of the soldiers in the unit, the formation, and the current effectiveness rating.

Although the documentation is a hefty 56 pages, *Decision in the Desert* is not hard to learn. Playing well takes practice, but an experienced computerist can skim the first two sections of the instructions and immediately start the conquest of North Africa. The rulebook is packed with examples of play and the kind of historical commentary which breathes life into a screenful of unit symbols.

If anything, *Decision in the Desert* is even better than the first title in this series, *Crusade in Europe.* Military simulation fans can only chortle in anticipation of what the Meier-Never team will create next.

MicroProse, 120 Lakefront Dr., Hunt Valley, MD 21030 (phone: 301-667-1151). *—Arnie Katz*

DAVE WINFIELD'S BATTER UP!

Avant-Garde
Commodore 64
Disk; $34.95

Warning: *Batter Up!* is not a baseball simulation, but rather a high quality tutorial on the subject of hitting a ball with a Louisville slugger. The teacher is, to say the least, highly qualified—New York Yankee superstar Dave Winfield.

Hitting a small round ball thrown at great velocity with the curving surface of a bat is universally acknowledged as the most difficult accomplishment in sports. *Batter Up!* divides this challenging feat into grip, stance, pitch, and swing. Drawings and animations minutely analyze the phase under discussion, and provide the pupil of hitting with many possible options.

To facilitate study of fastballs and changeups, the program shows the delivery from the batter's standpoint. Other pitches, the slider for example, are viewed from both overhead and side perspectives to allow the user to see how the ball breaks and drops at the same time.

The flexibility of this tutorial is impressive. The user picks the type of pitch, the delivery (overhand, submarine, etc.), the stance, and other factors by responding to a series of menu prompts.

An animation, which the user can run at regular speed or frame by frame, provides an overhead view of the swing. Simple illustrations focus on the proper way to grip the bat, while footprints indicate where the batter should plant his feet in the box.

After mastering the basics of the batsman's art, the student can try out newly acquired skills in two hitting simulations. In "Batting Practice," players may bat from either side of the plate, against a pitcher of their choosing. The action is in slow motion to give the user the chance to dissect each swing.

"Slugfest" pits the user against any one of a dozen pitchers, throwing real-time stuff. Options available in this version include updateable statistics and a choice of stadiums. The difference among the various fields seem minor, and the various parks have a negligible impact on the batter-pitcher confrontation.

"Batting Practice" and "Slugfest" depict the batter as seen from the on-deck circle. When the ball is hit, a window appears in the upper left corner of the screen with an overview of the field. The program tracks the path of the ball, and the result is displayed.

The obvious question is whether apprentice Winfields will really stand, bat in hand, in front of their Commodore computers and follow the master's teachings. Those who are willing to try such a radical method of instruction will probably derive some benefit from *Batter Up!* For safety's sake, however, it might be better for the computerist to use a whiffle bat rather than a major league stick...

Avant-Garde, 37B Commercial Blvd., Novato, CA 94947 (phone: 415-883-8083). *—Bill Kunkel*

NEXT MONTH'S ENTERTAINMENT REVIEWS:

- **Kung Fu: The Way of the Exploding Fist**
- **The Dolphin's Rune**
- **Richard Petty's Talladega**
- **Black Thunder**
- **Temple of Apshai Trilogy**
- **The Little Computer People Project**
- **Alice in Videoland**

TURNKEY 64: AN AUTO-EXEC CARTRIDGE FOR THE C-64

By Don Lewis

Due to its outstanding price/performance ratio the C-64 is finding its way into school and industrial laboratories as a dedicated computer system. As an example, Brachman Associates recently used a C-64 as a controller for an industrial process. The complete system consisted of a 64, a 1541 disk drive, printer, color monitor, and some custom circuitry. Most of the application program was written in BASIC and then compiled. The application called for the computer to begin executing the program immediately upon power up, without the presence on an operator to load the program from disk. A setup such as this is called a turnkey system.

The advantages of a turnkey system is that little or no operator intervention is required to get the program running. This means that anyone, with almost no instruction, can use the computer to control the process. With proper design, an inexpensive timer can be used to power up the system and automatically begin running the application program without any human intervention!

Many computer systems have a turnkey feature built into the operating system. Typically, a specific file name is reserved by the operating system for this purpose. When the computer is powered up, the operating system checks for the presence of this reserved file on the disk. If it is found, the file is loaded and executed. On the Apple, the reserved file is the HELLO file. On the IBM it is the AUTOEXEC.BAT file. Unfortunately, the C-64 does not have a turnkey feature similar to the Apple or IBM. However, such a feature can be easily added by placing the turnkey code in a ROM cartridge.

This article presents the code necessary to implement an AUTO-EXEC cartridge for the 64. Once the cartridge is installed on the 64, on power up or reset, the computer will load and begin executing the first file on the disk. Bulletin board systems, home security systems, and C-64-based science projects could all benefit from the addition of an AUTO-EXEC cartridge.

I originally developed this technique in 1983 for R.J. Brachman Associates, Inc., and I wish to thank them for permission to share it with *Ahoy!*'s readers.



Turnkey 64 uses readily available ROM cartridge with control flip-flop added.

SOME CARTRIDGE FUNDAMENTALS

Information on the C-64 memory configurations can be found on pages 260-267 of the *Programmer's Reference Guide.* A discussion of ROMs and a review of EPROM programmers for the C-64 can be found in the July 1985 *Ahoy!* The C-64 supports 8 different memory configurations, of which 4 are for cartridge ROM. The status of the EXROM and GAME lines (pins 8 and 9) of the expansion port determines the particular memory configuration the C-64 assumes on power up. When EXROM is tied low (to ground) the C-64 deselects the RAM at $8000-$9FFF and replaces it with the external ROM. This configuration was chosen for the AUTO-EXEC cartridge.

On power up or reset, the C-64 executes several housekeeping routines built into the Kernal ROM. One of these routines checks for the presence of a unique sequence of bytes at location $8000-$8008. If this unique sequence is present, the C-64 stops its normal reset routines and program control is transferred to the cartridge. Commodore calls this the "auto-start" feature.

Table of Autostart Recognition Code

Location	Definition
$8000, $8001	Low, High Byte of Cold Start Vector
$8002, $8003	Low, High Byte of Warm Start Vector
$8004-$8006	The ASCII sequence 'CBM', with bit 7=1
$8007-$8008	The ASCII sequence '80'

THE AUTO-EXEC CARTRIDGE

The AUTO-EXEC cartridge was designed to load and execute the first file on the disk. The necessary code is reproduced here, and provided in a *Flankspeed* listing on page 98. The comments explain how the code works. I built in several safeguards and a backdoor. If, while powering up the C-64, the CBM logo key is held down, the AUTO-EXEC is aborted and control returns to the normal C-64 reset procedures. Any disk errors, such as device not present or an error reading the file, also aborts the AUTO-EXEC. A more sophisticated AUTO-EXEC cartridge might write an error message to the screen and sound an alarm. The border, background, and character colors can all be changed according to individual taste.

The AUTO-EXEC cartridge assumes that the program it loads is a BASIC program or a machine language program started with a BASIC SYS call. The standard load address for such a file is at $0801.

This code can be burned into a 2732 EPROM with an EPROM burn-

er such as the Promenade. A schematic and instructions for building a burner yourself can be found in *Easy Interfacing Projects for the C-64* by Jim Downey, Don Rindsberg, and William Isherwood (Prentice-Hall). Figure 1 shows the necessary connections for interfacing a 2732 EPROM to the expansion bus. In order to map the cartridge ROM into the C-64 memory space, EXROM (pin 9 on the expansion connector) must be tied low (ground). A disadvantage of tying EXROM permanently low is that we give up 8K of RAM space which would otherwise be available to BASIC. With the addition of some extra circuitry the 8K of RAM can be recovered. With a little additional electronic wizardry we can effectively add 8K of ROM memory to the C-64.

Figures 2 and 3 show the method. By adding a 74LS74 D-type flip-flop as a one bit memory cell we can under software control enable the circuit in Figure 2 first. The operating rule for a D-type flip-flop is that the logic level at D is transferred to Q, whenever there is a high (+5 volts) to low (0 volts) transition on the CLK input. The PRE and CLR inputs override any previous condition. If a low is applied to CLR, the output of Q is immediately set low. A low on PRE sets Q high. When the C-64 RESET line is brought low during power up, Q is set low. This forces EXROM low and maps the external ROM into the C-64 memory space.

The I/01 line (pin 7) is a signal generated by the C-64 which is brought low whenever an address in the range $DE00-$DEFF is placed on the address bus. Commodore uses this line for their CP/M cartridge. In the AUTO-EXEC cartridge, it is used to switch the cartridge on and off. An instruction such as STA $DE00 (POKE 56832) will cause I/01 to go low. When I/01 is brought low, the high signal at D is transferred to Q and the ROM is disabled. In this circuit, once the ROM is disabled it cannot be reenabled.

The circuit in Figure 3 shows a more flexible arragement to wire the D flip-flop as a switch for the ROM. Instead of tying D permanently high (+5 volts), we connect D to the inverse of Q ($\overline{Q}$–read "not Q"). The first STA $DE00 would turn the cartridge

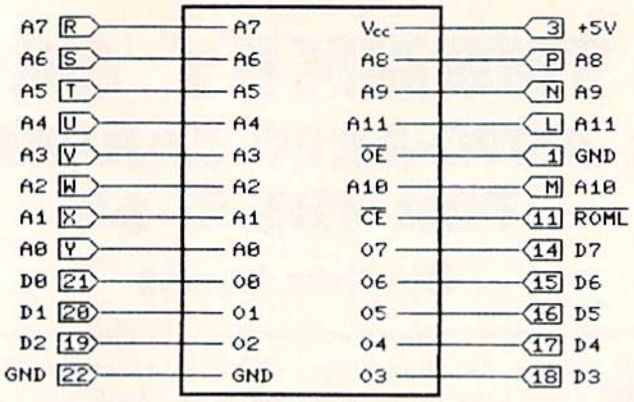


FIGURE 1:
2732 EPROM pin out. Pin designations for the C-64 expansion connector are in blocks. In addition, EXROM (pin 9) must either be tied low (ground), or wired as shown in Figure 2 or 3. See page 396, **Programmer's Reference Guide.**

SOURCE CODE LISTING

```
;AUTO-EXEC CARTRIDGE
;V1005/85 (RV 0606/85, 0831/83)
;CHANGE TO IO1 FOR ROM DISABLE

;COPYRIGHT 1983 BY
;R. J. BRACHMAN ASSOCIATES INC.
;P.O. BOX 1077, HAVERTOWN, PA. 19083
;PUT "0:AUTO-EXEC.A1005"

;64 KERNAL ROUTINES
IOINIT =$FF84 ;INIT IO
RAMTAS =$FF87 ;TEST RAM
RESTOR =$FF8A ;SET I/O VECTORS
CINIT  =$FF81 ;INIT SCREEN EDITOR
SETLFS =$FFBA ;SET LOGICAL FILE
SETNAM =$FFBD ;SET FILE NAME
LOAD   =$FFD5 ;LOAD RAM
CLALL  =$FFE7 ;CLOSE ALL FILES
SETMSG =$FF90 ;SET KERNAL MESSAGE
OUTCHR =$E719 ;OUTPUT CHAR TO SCREEN
LISTEN =$FFB1 ;COMMAND DISK TO LISTEN
SECOND =$FF93 ;SEND SA TO DISK
CIOUT  =$FFA8 ;CHAR TO DISK
UNLSN  =$FFAE ;COMMAND DISK TO UNLISTEN

;BASIC ROUTINES
BASICI =$E3BF ;INIT BASIC
BASVEC =$E453 ;SET BASIC VECTORS
CLEAR  =$A659 ;BASIC CLR COMMAND
CHAIN  =$A533 ;LINK BASIC LINES
DECTXT =$A533 ;BACKUP TEXT POINTER
RUN    =$A7AE ;GO TO IT!

;VARIABLES
TXTTAB =$2B ;START OF BASIC
VARTAB =$2D ;START OF VARIABLES
EAL    =$AE ;LOAD END ADDRESS
STATUS =$90 ;I/O STATUS
MEMSZ1 =$37 ;HIGHEST ADDRESS FOR BASIC
MEMSZ2 =$0283 ;TOP OF MEMORY FOR O.S.
COLOR  =$0286 ;CURRENT CHAR COLOR
TBUFFR =$033C ;TAPE BUFFER
ISTOP  =$0328 ;STOP ROUTINE VECTOR

;64 I/O
VICCTR =$D016 ;VIC CONTROL REGISTER
BORDER =$D020 ;BORDER COLOR
KBPROB =$DC00 ;PROBE THE KEYBOARD
KBSENS =$DC01 ;SENSE THE KEYBOARD
IO1    =$DE00 ;EXTERNAL I/O SELECT

;CIA #1 TIMER REGISTERS
CIATA  =$DD04 ;TIMER A
CIAICR =$DD0D ;IRQ CONTROL REGISTER
CIACRA =$DD0E ;CONTROL REGISTER A

 *=$8000
       .WORD COLD

;SET NMI(RESTORE KEY)
;FOR REGULAR ENTRY POINT
       .WORD $FE5E

;THE AUTO-START SEQUENCE
       .BYTE $C3,$C2,$CD,'80' ;CBM80

;X=0 ON ENTRY
COLD   STX VICCTR
       JSR IOINIT
       JSR RAMTAS
       JSR RESTOR
       JSR CINIT
;SET BORDER, BACKGROUND & TEXT COLOR
       LDA #12
       STA BORDER
       LDA #0
       STA BORDER+1
       LDA #14
       STA COLOR

;CHECK KEYBOARD FOR CBM KEY
       LDA #$7F
       STA KBPROB
       LDA KBSENS
       CMP #$DF
       BNE COLD1 ;NOT CBM

;CBM KEY, ABORT AUTO-EXEC
BASIC  LDX #32
BASIC1 LDA CODE1,X
       STA TBUFFR,X
       DEX
       BPL BASIC1
       JMP TBUFFR

COLD1  JSR BASVEC
       JSR BASICI

;GIVE DEVICES ON BUS TIME TO
;FINISH POWER UP SEQUENCE
       LDX #32

;SET UP TIMER 1, CIA 1
TIME   LDA #$FF
       STA CIATA
       STA CIATA+1
       LDA #$19  ;FORCE LOAD
       STA CIACRA
```

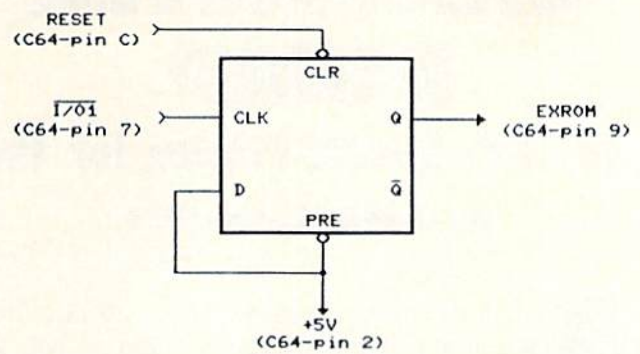


FIGURE 2:
A 74LS74 D-type flip-flop can be used to turn off the cartridge ROM, thereby recovering 8K of RAM which would be hidden behind it.

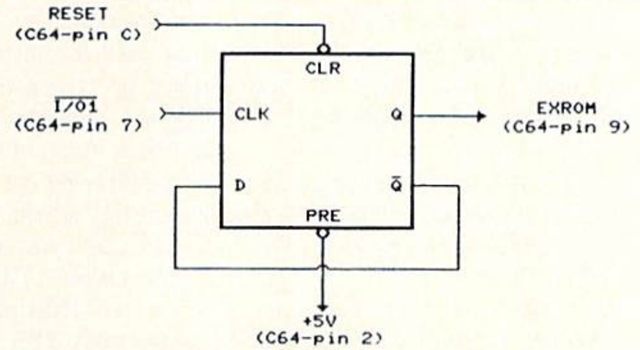


FIGURE 3:
When D and Q are wired together, the flip-flop is configured as a toggle. This allows the cartridge ROM to be alternated with underlying RAM in the C-64 address space under program control.

off, a second store would turn it back on a third store would turn it off again, and so on. By doing this, we have effectively added 8K of ROM memory to the C-64. A library of machine language subroutines, perhaps for high speed graphics, drivers for analog to digital convertors, or high speed disk access, could be placed in the external ROM. A BASIC application program could access these routines by first turning the ROM on with a PEEK(56832), calling the desired routine with a SYS command, then turnign off the ROM with another PEEK(56832).

If you are not up to building your own EPROM cartridge board then use one of the reasonably priced prepared boards available from the EPROM programmer manufactures. The photograph on page 53 shows such a board with a cheap and dirty way of installing the 74LS74 flip-flop. You may want to use a socket instead of soldering directly to the chip.

Additional information on digital electronics and microcomputer systems can be found in *Engineer's Notebook II* by Forest M. Mims III (Radio Shack), *Digital Systems* by Ronald J.

FOR TURNKEY 64

```
       LDA CIAICR ;CLEAR PENDING IRQS
;WAIT FOR TIMER TO COUNT DOWN
TIMOUT LDA CIAICR
       AND #$01
       BEQ TIMOUT
       DEX
       BNE TIME

;TEST IF DEVICE PRESENT
       LDA #0
       STA STATUS
       LDA #8
       JSR LISTEN
       LDA #$6F
       JSR SECOND
       LDA STATUS
       BEQ COLD2

;NO DEVICE ON BUS, JUST EXIT
ERROR  LDA #0
       STA STATUS
       JMP BASIC

COLD2  JSR UNLSN

;LOAD IN FILE, ASSUMES A BASIC PROGRAM
;OR ML PROGRAM STARTED WITH A SYS CALL.
       LDA #$EA ;FIRST DISABLE STOP
       STA ISTOP
       LDA #8   ;FILE #8
       TAX      ;DEVICE #8
       LDY #0   ;LOAD, NOT VERIFY
       JSR SETLFS
       LDA #4   ;FILE NAME LENGTH
       LDX #<NAME ;POINTER TO FILE NAME
       LDY #>NAME
       JSR SETNAM
       LDA #0       ;RELOCATED LOAD
       LDX TXTTAB ;AT $0801
       LDY TXTTAB+1
       JSR LOAD    ;AND LOAD THE RAM
       BCS ERROR
;CARRY CLEAR, NO ERROR
       JSR CLALL

;SET END OF PROGRAM POINTERS
       LDA EAL
       LDY EAL+1
       STA VARTAB
       STY VARTAB+1

;CLEAR BASIC VARIABLE SPACE
;AND RELINK BASIC LINES
       JSR CLEAR
       JSR CHAIN
       LDA #0   ;TURN OFF MESSAGES
       JSR SETMSG
       JSR DECTXT ;ADJUST TEXT POINTERS

;MOVE AND EXECUTE CODE
       LDX #32
EXEC   LDA CODE2,X
       STA TBUFFR,X
       DEX
       BPL EXEC
       JMP TBUFFR

;FILE NAME, IN THIS CASE LOAD IN
;FIRST FILE ON DISK
NAME   .BYTE '0:?*'
;ERROR HANDLER, CODE MOVED TO
;TAPE BUFFER & EXECUTED THERE
CODE1  STA IO1    ;CARTRIDGE OFF
       LDA #00    ;TEST IF ROM OFF
       STA $8004
       BIT $8004
       BMI CODE1A ;NOT RAM
;IS RAM, SO RESET
;TOP OF MEMORY TO $A000
       LDY #$A0
       STA MEMSZ1
       STA MEMSZ2
       STY MEMSZ1+1
       STY MEMSZ2+1
CODE1A LDA #$ED
       STA ISTOP ;ENABLE STOP
       JMP ($A000) ;AND
                   ;GOTO BASIC(READY.)

;EXECUTE PROGRAM, CODE MOVED TO
;TAPE BUFFER & EXECUTED THERE
CODE2  STA IO1    ;CARTRIDGE OFF
       LDA #00    ;TEST IF ROM OFF
       STA $8004
       BIT $8004
       BMI CODE2A ;NOT RAM
;IS RAM, SO RESET
;TOP OF MEMORY TO $A000
       LDY #$A0
       STA MEMSZ1
       STA MEMSZ2
       STY MEMSZ1+1
       STY MEMSZ2+1
CODE2A LDA #$ED
       STA ISTOP ;ENABLE STOP
       JMP RUN   ;AND RUN PROGRAM

       .END
```

Tocci (Prentice-Hall), and *Digital Integrated Circuits* by Joseph Kasper (Prentice-Hall). The last book is highly recommended for beginners.

Special thanks to Bruce I. Brachman and to the management of R.J. Brachman Associates, Inc. The illustrations were created with *Flexidraw* from Inkwell systems. □

SEE PROGRAM LISTING ON PAGE 115

Instructions for Testing the AUTO-EXEC Cartridge

1. The prototype works best if your C-64 and disk drive are powered off the same switch on a power strip. Otherwise you will need to turn the drive on first, then power up the C-64.

2. Insert the cartridge in the expansion port, and insert a disk with a suitable boot file in the drive. Close the drive door.

3. Power up the C-64. The screen will be black with a gray border. After a slight delay, the C-64 will begin to load the first file off the disk and then begin executing that file.

4. Test the backdoor and safeguard features by powering up the C-64 with the drive turned off. The C-64 will power up with the normal startup message. The same thing will happen if the drive is on but no disk is present. Test the backdoor by powering up while holding the CBM logo key down.

5. Test the ROM disable feature as follows: once the boot program has finished its job, and the READY message appears, type in direct mode POKE 32768,0:PRINT PEEK (32768). A 0 will be printed, showing that RAM is present at $8000 (32768). Now type a PRINT PEEK (56832). This will bank the ROM back into the C-64 memory space. Try to POKE to 32768. You will not be able to change the value there since it is ROM. Type PRINT PEEK (56832) a second time. This will bank the ROM out and the RAM back in. Try a POKE to 32768 now. You will now be able to change the value.

CHARACTER DUMP

Instant Text Screen Printer for the 64

By James C. Sanders

Many versions of screen dump programs have been devised since the home computer was introduced, with a variety of shortcomings. A screen dump program written in BASIC must be added to another program in order to load properly and work with that program. Many dump programs will only work with one printer. A dump program in machine language, however, will be free of any interference with the BASIC program you may be working with.

The first version of *Character Dump* was written with only 97 bytes of memory. It proved very valuable, but had to be activated with a system command from either the program or by direct mode.

It became apparent that a screen dump activated by a function key would be much more useful. The program was rewritten to use the f7 key to activate the procedure. The program is now 149 bytes of machine language memory. It has been located at a high memory address of 52992 ($CF00) to prevent interference with other ML programs located at the 49152 (C000) location.

This version of the program is presented with a BASIC loader. This loader, when run, will load to the memory address 52992 and then erase the loader from memory. It will print the checksum value of all the data statements and instructions to the screen befoire doing so.

You may save the instructions to your printer by typing SYS 52992 <RETURN> and pressing the f7 key. A different function key may be selected by following the POKE instructions on the screen. It will revert to key f7 when you run the program the next time. The program may also be activated from within a BASIC program by SYS 53034. Direct mode entry of the SYS 53034 command will also activate the program.

The program will work a wide variety of printers, including all Commodore, Silver Reed, Olivetti, Sanyo, and Panasonic printers. Remember, it is not a graphics or high resolution screen dump, but one for note-taking as you program. Some game cartridges and fast load programs will interfere with the proper execution of the program. Remove them prior to loading *Character Dump.* □

SEE PROGRAM LISTING ON PAGE 118

REVIEWS

S'MORE

Cardco, Inc.
Commodore 64
Cartridge; $69.95

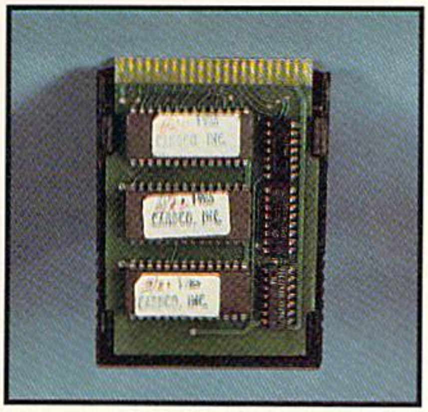
S'MORE is Cardco's third offering in their catchy name product line. The first two were the G-Wiz and its prototype Oui-G printer interfaces. We can't give them all the credit for dreaming up this one, as a survey of your grocer's shelves will readily show. Nevertheless, Cardco gets full credit for the Super Memory Optimized RAM/ROM Expansion they devised for their chosen acronym. In any event the name is entirely apropos, as a *S'MORE* cartridge in a C-64 will definitely provide more of all those things that BASIC programmers are looking for.

Some of what's more is immediately apparent from the moment you power up. The dark gray on white opening screen shows 61,183 bytes free, more than 1½ times what is normally available to C-64 BASIC 2.0. As if to emphasize the point, BASIC's "READY" prompt has also been replaced by "ok". Perhaps the *S'MORE* operating code was originally written in FORTH. The extra RAM is not built into the *S'MORE* cartridge. It is part of the original 64 kilobytes of RAM which comes with the computer. *S'MORE* makes this memory available to the BASIC programmer by adding memory handling hardware and firmware similar to that which is built into the Commodore Plus/4 computer.

The *S'MORE* cartridge does contain 24 kilobytes of ROM, which supplements the 16 kilobytes already in the C-64. This additional code does a lot more than manipulate the C-64 RAM. *S'MORE* adds some 47 much-needed commands to BASIC 2.0. In addition, many of BASIC's existing commands are enhanced in subtle and convenient ways.

Many of the added commands are in the Programmer's Aid and Operating System Support categories. Heading the list are the AUTO and NUMBER commands which take the tedium out of numbering and renumbering BASIC program statements. The number command will also flag unresolved line numbers in GOTOs and GOSUBs by assigning them to line number 63999. A rare treat for Commodore programmers are the FIND and CHANGE commands which locate and substitute text strings anywhere in your program. A useful application is to FIND the aforementioned 63999s.

With *S'MORE* the ability to create and maintain BASIC subroutine libraries is more than possible. In fact it becomes downright convenient to organize your program files and utilities. The DELETE command, which provides for the elimination of program lines en masse, supports easy creation of program modules. The MERGE command will recombine these modules with any other program in memory. This is a true merge where the added lines are properly placed into the existing program's line number sequence. Incoming lines will also replace existing lines with the same numbers, so watch out. Liberal use of the NUMBER command will of course help keep things straight.





Cardco's S'MORE *gives you s'more ROMs—three 8-kilobyte ROMs. READER SERVICE NO. 155*

When MERGEing programs be sure to bring in the longest ones first. The MERGE operation is time consuming. Combining a long program with a short one in memory can take several minutes—after the disk drive has stopped working.

Life with the 1541 disk drive is greatly improved by *S'MORE*. A PRINT DS$ will read and display the disk error channel. The CATALOG command displays the directory without affecting the program in memory. LOAD may be typed right over a directory display as the default device is now the disk drive instead of the cassette. The trailing PRG in the display is effectively ignored as well. Programs may be LOADed and RUN in one fell swoop by the RUN"prg name" command, and the Commodore/RUN STOP keys now LOAD and RUN the first program on the disk. The new DISK command facilitates sending any of Commodore's DOS commands to the disk drive. In all, the added disk support is nearly as convenient as the DOS wedge.

The function keys are easily programmed with *S'MORE*. The KEY command will assign any string (up to a combined total of 128 bytes) to the function keys. Some useful commands are placed in the function keys by *S'MORE* on power up. For example, f3 will read the disk drive error channel while f7 displays the directory.

S'MORE includes several facilities for program debugging. A running program may be STOPped at any

time, its non-array variables DUMPed, and CONTinued from where it was interrupted. The TRACE command can be used to keep track of the program lines as they execute. The TRAP and RESUME commands will even intercept BASIC programming errors without aborting the program. The ER, EL, and ERR$ variables let you identify the specific errors and their line numbers as they are TRAPed. If you should encounter a syntax error, the HELP command will show you the line where it's located.

The commands for inputting and outputting data are enhanced by *S'MORE*. The AT command locates the cursor on the display screen. The USING as in PRINT USING and the associated PUDEF commands conveniently format printed output. Control of user input is improved with the INLINE (accepts punctuation and quotes), INFORM (defines the input field), and GETKEY (waits for a keypress) commands.

S'MORE comes with a detailed 150-page manual which is up to Cardco's usual high standards. Each command is thoroughly documented, complete with illustrative examples.

A *S'MORE* system memory map is included to let you adjust machine language subroutines for the *S'MORE* environment. The Commodore Kernal vectors remain unchanged, as well as many key locations in low RAM. However, most other machine-dependent locations have been significantly modified. For one thing, the C-64's "protected" RAM in the $C000 block is now part of *S'MORE* BASIC's workspace. Also, all of the I/O block is no longer directly accessible by PEEKing and POKEing. Instead, *S'MORE* BASIC provides a complete set of reserved variables for this purpose. Even the screen memory is no longer in the same place.

The manual does mention that the *S'MORE* memory banking is similar to what is used by the Commodore Plus/4. It even refers you to the Commodore documentation for a description. Unfortunately, the Plus/4 technical notes are not readily available to the general public. We would have liked to see

s'more details on the *S'MORE* memory banking in its manual.

A *S'MORE* demo disk with numerous illustrative programs is included in the package. Since the disk is not part of the advertising, Cardco does not feel the need to cover it under warranty. They will replace it for $3 if it fails to work.

Although *S'MORE* is fully upward compatible with BASIC 2.0, any programs using PEEKs and POKEs will probably require modification. From the standpoint of using existing programs, it will be best to treat a C-64 with *S'MORE* as a new Commodore computer.

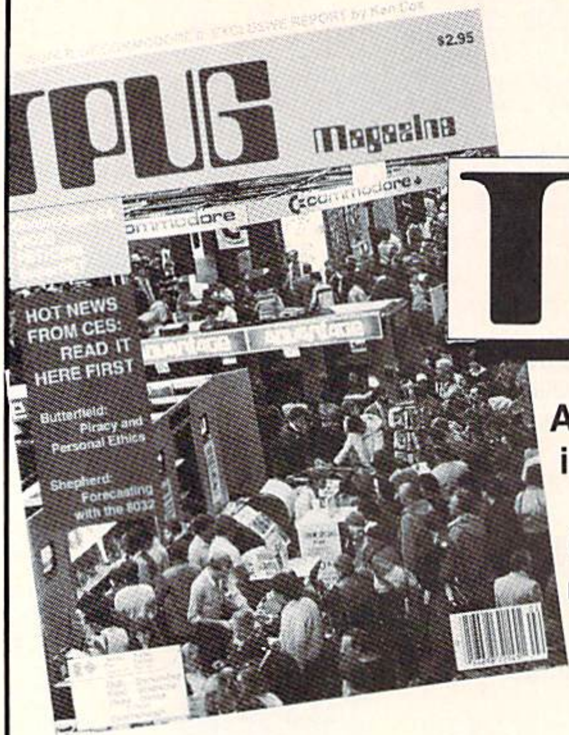
As with other cartridge firmware for the C-64, *S'MORE* automatically asserts itself on power up. *S'MORE* may be incompatible with some early versions of the C-64. If your display fills with commercial at (@) symbols you will know what we mean. *S'MORE* will also work the C-128 in C-64 mode if the Commodore logo key is held down when powering up or when the reset button is pressed. Turning on the C-128 with *S'MORE* in place and without following the above sequence prevents the computer from initializing. The GO64 command will not work at all if *S'MORE* is in the expansion port.

You should also be able to use *S'MORE*'s MERGE facility with C-128 programs. However, programs will not LIST properly, as the C-128's BASIC 7.0 tokens are not compatible with *S'MORE* BASIC.

After all the good things we found in *S'MORE*, we had to really dig deep to come up with some shortcomings. The complex memory management performed by *S'MORE* does eat up some processor time. Some BASIC operations, such as DIMensioning and filling large arrays or performing many repetitive calculations, will run a bit slower than on the C-64. The penalties are not severe and may be more than offset by *S'MORE*'s improved garbage collection routines. The latter are nearly 30 times faster than in BASIC 2.0.

If you are really after speed, Cardco is planning to offer a *S'MORE* compiler in the near future. The com-

THE TORONTO PET USERS GROUP INC.



TPUG

A membership in the world's largest Commodore users' group will provide you with:



- 10 issues of TPUG magazine
- Advice from experts like Jim Butterfield and Elizabeth Deal
- Access to a huge library of outstanding public domain software
- An invitation to attend the Annual TPUG conference, May 25 and 26 in Toronto, Canada, with two full days of seminars on all aspects of Commodore computing.

JOIN NOW!

I would like to join TPUG as an associate member at a cost of US$25.00.

Send your cheque, money order or credit card number to:

TPUG Inc.
101 Duncan Mill Road, Suite G7
Don Mills, Ontario, Canada
M3B 1Z3

Name ______________________________

Address ______________________________

____

☐ Cheque ☐ Money Order ☐ Credit Card __________ Amount in U.S. funds $ ________

Credit Card # __________ Expiry date __________ Signature __________

Type of Computer ________ Disk Drive ________ Printer ________ Modem ________ Other ________

piler will require a *S'MORE* cartridge.

The following table summarizes the results of our benchmark tests:

S'MORE **BENCHMARK TESTS**

	CALC	DIM	GARBAGE
S'MORE	8238	3527	45
C-64	6158	2586	1296
C-128	7941	3442	12
C-128 (FAST)	3758	1627	6

Note: all times are in jiffies (1/60 second).

Since *S'MORE* was so convenient to have around, we didn't want to unplug it from the computer. However, this was often necessary when running commercial software or existing utilities. The *S'MORE* modifications to the C-64 memory map are not compatible with most existing programs. A useful addition to the *S'MORE* cartridge would be a simple way to enable and disable it without having to unplug it. A cartridge expander board, with its own control switches, may not be the solution, due to the space limitations of many installations.

We have presented here just some of the features of *S'MORE*. In fact the more we use *S'MORE* the more we find out about it and the more we like it. If you are still looking for a BASIC support utility for your C-64 then *S'MORE* certainly merits serious consideration.

Cardco, Inc., 300 S. Topeka, Wichita, KS 67202 (phone: 316-267-3807). *—Morton Kevelson*

TABLE OF S'MORE BASIC KEYWORDS

Group 1 – Programmer's Aid

HELP	– Identifies syntax error
	– Same as REM
AUTO	– Automatic line numbering
*LIST	– Does not terminate program
DELETE	– Deletes program lines
FIND	– Locates text in program
CHANGE	– Replaces found characters
NUMBER	– Renumbers program lines
DUMP	– Lists non-array variables
TRACE	– Tracks program execution
OLD	– UnNEW
DEC	– Hexadecimal to decimal
HEX$	– Decimal to hexadecimal
KEY	– Assigns function keys
TRAP	– Where to go on syntax error
ER, EL	– Error number and line
ERR$	– Error message
RESUME	– Where to go after error

Group 2 – Disk Support

CATALOG	– Displays disk directory
*LOAD	– Defaults to device 8
*RUN	– Loads and RUNs program
MERGE	– Merges BASIC program
DISK	– Sends DOS command
DS, DS$	– Reads DOS error channel
DOPEN#	– OPENs relative file
RECORD#	– Accesses relative file record

Group 3 – User Interface

AT	– Locates cursor as in PRINT AT
USING	– Prints with format as in PRINT USING
PUDEF	– Defines format for USING
*INPUT	– Works with AT
INLINE	– INPUT accepts quote, colon and comma. Accepts only string variables. Works with AT
INLINE#	– Same as INLINE for file#
INFORM	– Formatted INPUT
GETKEY	– Waits for keypress

Group 4 – Structured Programming

DO/LOOP	– Start and end of do loop
WHILE	– Use in do loop
UNTIL	– Ditto
EXIT	– Leave do loop now
ELSE	– As in IF...THEN...ELSE

Group 5 – PEEK and POKE Eliminators

CIA(x)	– The CIA chips
COL(x)	– Color memory
SID(x)	– The SID chip
VIC(x)	– The VIC chip
VID(x)	– Screen memory

Note: The parameter in each of these reserved variables corresponds to a register or memory address.

Group 6 – Odds and Ends

INSTR	– Finds substring within string
*MID$=$	– Permits substitution of string
*RESTORE	– Resets pointer to specified line
CLS	– Clears screen
UPPER	– Sets upper case/graphics
LOWER	– Sets upper/lower case
REPEAT	– Controls key stutter
NORM	– Clears screen and resets colors to default conditions
*STOP	– Enables or disables the STOP key
RESET	– System reset
MONITOR	– Go to machine language monitor
BORDER	– Sets border color
PAPER	– Sets screen color
INK	– Sets character color

*enhanced version of existing command

NLQ KIT FOR GEMINI 10X

ESP Corporation
Price: $57.50

Psstt!

Say, how about a new Gemini 10X printer for $57.50? It's got a typeface that's so near letter quality, you have to see it to appreciate it.

Naw, it's not hot merchandise! Look at this—the 'y' doesn't look like a 'v' with one side stuck below the line. This printer makes the 'y' look like a 'u' with a curved tail. Look at the 'p' on this $57.50 printer—it's got serifs on it. The letters are round, not square. Easier to read.

Where can you get a printer like this? If you have a Gemini 10X, you already own one. But for $57.50, a module about half the size of a cassette tape is available that will upgrade the typeface. The German-made module has been used in the European Gemini models for about a year. Only recently did it become available in the United States.

Just about anyone can install it. The printer comes apart by removing two screws in the rear of the top deck. The module replaces the two integrated circuits sitting side-by-side in the center of a circuit board located directly behind the carriage. It's just a matter of prying them out of their sockets (a small jeweler's screwdriver comes in handy to get them worked loose).

Once out, the near-letter-quality module plugs in the sockets. In 20 minutes, the chore is done.

The NLQ module works its magic by the way it manipulates the dot-matrix pins and moves the printhead. The letters are formed during two swipes across the paper. On the first left-to-right pass, nearly all of each letter is printed, except for a tiny bit on the bottom. The 'y', for instance, has no descender. On the next left-to-right pass, the strip on the bottom is filled in tightly, so it is impossible to detect the printhead's two-step method in the finished product.

Besides having to do the installation yourself, there are some other tradeoffs for getting print quality as good as Star Micronics' newer SG-10 for a fraction of the price.

Printing a one-page letter with 230 words took 43 seconds using the Gemini in its normal, emphasized mode. With the NLQ chip turned on, the same letter took 83 seconds. Still, that's an average of 14 characters a second, or about the speed of some inexpensive daisy wheel printers.

The near letter quality typeface resides where the italic face used to be, so you'll sacrifice the slanted letters. But the new type provides sufficient consolation. And the NLQ mode can be used with expanded and underlined words.

The letter forms are pleasing, with the exception of one slight flaw: the double quotation marks and the single apostrophe curve ever-so-slightly toward the left, making them look a little out of place at the start of a sentence or phrase. They'd have been perfect if left straight.

```
This is the Gemini
10x printer in the
draft mode.
```

This is the Gemini
10x printer in the
emphasized mode.

This is the Gemini
10x printer using the
near-letter-quality
module.

I've tried the NLQ chip with some popular word processing programs, including *Fleet System 2*, *PaperClip*, and *Easy Script*. It worked every time. Nor did it interfere with screen dumps or offerings from *The Print Shop*.

It would be a valid comparison to say that the NLQ module has done for the Gemini 10X's typeface what the fast loading cartridges have done for the 1541 disk drive's speed.

The NLQ Kit works with the Gemini 10X and 15X and the Delta 10 and 15 printers.

ESP Corporation, 7900 N. Tamiami Trail, Sarasota, FL 34243 (phone: 813-355-6797). *—Lonnie Brown*

REVIEWS

THE COMMUNICATION EDGE
THE MANAGEMENT EDGE
THE NEGOTIATION EDGE
THE SALES EDGE
Info Designs, Inc.
Commodore 64
Disk; $79.95 each

Info Designs' *Edge* series of software for Commodore 64, 128, and Plus/4 includes *The Communication Edge*, *The Management Edge*, *The Negotiation Edge*, and *The Sales Edge*. As the names imply, the purpose of these programs is to provide you with a strong base in your interactions with other people.

According to Info Designs, *The Communication Edge* "can help you evaluate the personalities, strengths, and weaknesses of people in your meetings, and provide specific tactics to listen accurately and speak persuasively." *The Management Edge* "can help you increase motivation and prevent manager/staff conflicts." *The Negotiation Edge* "can help you anticipate likely opponent maneuvers, implement alternative tactics and test their effectiveness." *The Sales Edge* "can help you maximize sales calls by providing insights into customer buying styles."

These programs perform a sort of online psychological assessment of you and of the individual with whom you will be interacting. Actually, they only assess your impressions of that other person. I guess you could have that person enter his own data, but then you would lose the edge the software is supposed to give you.

Each program asks you to agree or disagree with a number of statements or sentences about yourself and a number of single-word characteristics about your counterpart. You might agree that, for you, "fair play is a losing strategy" and you may disagree that your counterpart is either "honest" or "egotistical." The statements and traits are not organized in any obvious way and fluctuate from positive to negative.

You will want to complete the self-assessment first, which will take you about 20 minutes. Each program will display 86 to 90 statements on the screen, one at a time, except *Nego-*



Back Issues

Try this experiment! Flip through any issue of *Ahoy!* that you've finished reading. Add up the hours of programming toil our feature articles and tutorials saved you—the hours of shopping our expansive news section saved you—and the money our unbiased reviews saved you—and see if the magazine didn't pay for itself! Then ask yourself: can you afford to be without a single time- and money-saving back issue?

If you're serious about programming, you can't be serious about continuing without a complete collection of *Ahoy!*

ISSUE #1—JAN. '84 $4.00
64 v. the Peanut! Telecommunications! Tales of the Commodore! And ready to enter: Multi Draw 64 Graphics System! Interrupt Music Maker/Editor! A Peek at Memory! Sequential Files!

ISSUE #2—FEB. '84 $4.00
Illustrated tour of the 1541! Artificial intelligence! Synapse's Ihor Wolosenko interviewed! And ready to enter: Music Maker Part II! Screen Manipulation! Night Attack! Relative Files!

ISSUE #3—MAR. '84 $4.00
Anatomy of the 64! Printer interfacing for VIC & 64! Educational software series begins! And ready to enter: Address Book! Space Lanes! Random Files on the 64! Dynamic Power!

ISSUE #5—MAY '84 $4.00
Future of Commodore! Inside BASIC storage! Memory management on the VIC & 64! Guide to spreadsheets! And ready to enter: Math Master! Air Assault! Biorhythms! VIC Calculator!

ISSUE #6—JUNE '84 $4.00
Game programming column begins! Program generators! Rupert on inputting! Memory management continues! And ready to enter: Post Time for the 64 & VIC! Alpiner! Sound Concept!

ISSUE #7—JULY '84 $4.00
MSD dual disk drive! Database buyer's guide! Training your cursor! Screen displays! File Sleuth! Users Groups! And ready to enter: Renumbering! Checklist! Math Defender! Brisk!

Sorry...issues #1, 2, 4, and 7 sold out! Reprint editions (programs and feature articles only) of issues #1, 2, and 7 *only* are available for $4.00 each.

ISSUE #8—AUG. '84 $4.00
Choosing a word processor! Computational wizardry! Creating your own word games! Sound on the 64! And ready to enter: Micro-Minder! Directory Assistance! The Terrible Twins!

ISSUE #9—SEPT. '84 $4.00
Program your own text adventure! Build a C-64 cassette interface! Video RAM! And ready to enter: Salvage Diver! DOS! Sound Explorer! The Castle of Darkness! Base Conversions!

ISSUE #10—OCT. '84 $4.00
C-64 graphics programs! Bit-mapped graphics! Joystick programming! And ready to enter: VIC 40 Column Operating System! BAM Read & Print! Emerald Elephant! Lawn Job!

ISSUE #11—NOV. '84 $4.00
Music programs & keyboards for the 64! Graphics feature continues! And ready to enter: PTE word processor! Block Editor! Alternate Character Set for the 64! The Tunnel of Tomachon!

ISSUE #12—DEC. '84 $4.00
Buyer's guide to printers! 1525 printer tutorial! Custom characters! User Guide to KMMM Pascal! Diving into BASIC! And ready to enter: Construction Co.! Space Patrol! Cross Ref!

ISSUE #13—JAN. '85 $4.00
VIC/64 OS exposed! Sprites! 1541 device # disconnect switch! Ghostbusters! And ready to enter: Ultra Mail! Music Tutor! Alice in Adventureland! Midprint! To the Top! Tape/Disk Transfer!

ISSUE #14—FEB. '85 $4.00
Printer interfacing! Multicolor sprites! Modems! Bulletin boards! Theory of game design! And ready to enter: Futurewar! Fontasia! VIC Eraser! Insurance Agent! Flankspeed! Telelink 64!

ISSUE #15—MAR. '85 $4.00
Making multiscreen gameboards! Inside the Plus/4! Commodore DOS! And ready to enter: Old Routine! Programmable Functions! Automatic Line Nos.! Home Budget! Salmon Run!

ISSUE #16—APR. '85 $4.00
Assembly language column begins! Programming the joystick! 1541 disk drive alternatives! And ready to enter: Hop Around! Faster 64! Booter! Elecheck! BASIC Trace! Space Hunt!

ISSUE #17—MAY '85 $4.00
Disk drive enhancements! Install a reset switch! Assembler escapades! And ready to enter: Super Duper! Two-Column Directory! DSKDU! Raid! DOS Plus! Font Editor! Tile Time!

ISSUE #18—JUNE '85 $4.00
Music & graphics entry systems! How modems work! Inside the 6510! And ready to enter: Quad-print! Mapping 4.4! Towers of Hanoi! Speedy! Duck Shoot! The 6510 Simulator!

ISSUE #19—JULY '85 $4.00
PROM programming! 3-part harmonies on VIC/64! Speeding pixels! And ready to enter: Auto-Append! Script Analysis! Wizard of Im! Lucky Lottery! Brainframe! Etch! Printat!

ISSUE #20—AUG. '85 $4.00
Inside the 128! Read-world simulations! Sound effects! And ready to enter: Windows! Formatter! Sound-a-Rama! Screen Dump! Selectachrome! Disintegrator! Fidgits! Gators N Snakes!

ISSUE #21—SEP. '85 $4.00
Inside the 1571 drive and 128 keyboard! Sprite programming! And ready to enter: Fastnew! Go-Lister! File Lock! Dragon Type! Superhero! Auto-Gen! Moxey's Porch! Fish Math!

ISSUE #22—OCT. '85 $4.00
Create cartoon characters! Infinitesimal intrigue! Inside copy protection! And ready to enter: Shotgun! Maestro! Solitaire! Mystery at Mycroft Mews! Gravinauts! 1541 Cleaning Utility! ShadeyDump!

ISSUE #23—NOV. '85 $4.00
Adventure gaming! ML sprite manipulation! BASIC for beginners! And ready to enter: Lightning Loader! Knight's Tour! Chopper Flight! Rhythmic Bits! Instant Bug Repellent! File Scout! Slither!

ISSUE #24—DEC. '85 $4.00
Speech synthesizers! The IBM Connection! The year's 25 best entertainments! And ready to enter: Gypsy Starship! Directory Manipulator! Cloak! Gameloader! Jewel Quest! Lineout! Santa's Busy Day!

ISSUE #25—JAN. '85 $4.00
Build a speech synthesizer! Survey of sports games! And ready to enter: The Martian Monsters! Streamer Font! Microsim! The Haunted Castle! Knockout! Infraraid! Alarm Clock! Memory Check!

Use coupon or facsimile. If ordering more than three issues, list choices on separate sheet.

If you have a modem and want more complete information on any of the back issues listed, call *Ahoy!*'s Bulletin Board Service at 718-383-8909.

Ahoy!
BACK ISSUES
ION INTERNATIONAL INC.
45 West 34th Street
Suite 407
New York, NY 10001

Please Send Me The Following:
_____ Copies of issue number _____
_____ Copies of issue number _____
_____ Copies of issue number _____

Enclosed Please Find My Check or Money Order for $_____
(Outside the USA please add $1.00 for every copy)

NAME_____
ADDRESS_____
CITY_____ STATE_____ ZIP CODE_____

tiation, which displays 5 per screen. Your options are to agree, disagree, go to the next statement, or go to the previous statement. Or you can quit the self-assessment, save what you have entered, and return to the main menu. From program to program, the assessments bear a lot of similarity to one another.

Next, you assess your counterpart (in *Communication* and *Negotiation*), customer (in *Sales*), or subordinate (in *Management*). *Negotiation* gives you the most thorough opportunity for this assessment, asking you to respond to 90 traits and 22 statements about your counterpart. *Communication* and *Management* only ask you about 55 traits, and *Sales*, 66. In each case, 10 traits are listed on the screen; when you respond to the last one, the next screen appears. It is easy to visually verify your response to every trait except the last one on each screen. The screens change so fast that you will have to back up, with the "previous" command, if you are afraid you hit A (agree) instead of D (disagree), or vice versa.

In assessing your counterpart, you have the same options as you did in the self-assessment. Each of the programs except *Negotiation* also offers some level of help. *Communication* and *Management* "help" you by reminding you that each question requires a response and by describing the responses—agree, disagree, next, previous, and quit. *Sales* has far and away the best help function for assessment of your customer. Hit the "H" key and the screen will display a definition of the trait to which you are responding as well as your possible responses. All four programs should have this quality of online help.

Before you can print a strategy report, you must have responded to every trait in both the self- and counterpart assessments. Assessments are stored directly on the program disk. You are limited to storing one self-assessment and nine assessments of others. If two people use the program, their self-assessments will have to be completely reentered each time they switch or they will each have to copy the sequential file containing their data to another disk and then back to the program disk. The program disk is heavily copy protected, so expect to do a lot of writing to it. "What, me worry?"

After completing the assessments, choose to print a strategy report. The program will prompt you to remove the program disk and insert the output disk that comes in the package. The output disk stores "canned" paragraphs that the program cuts and pastes into a final report based on its analysis of your input. All the programs, except *Management,* use both sides of this second disk for text, so there is a fair amount of material from which the program may select.

Communication and *Management* reports will be broken into sections with headings like "Be Flexible" or "Create Moderate Stress." *Negotiation*'s report is organized into a summary of your counterpart's expected attitudes, a description of his initial position, tactics for you to consider, and a game plan suggesting possible openings, in-process, and closing arguments. The report from *Sales* tells you what to expect from your customer's style compared to yours, lists steps to succeed with your customer, and gives you opening, presentation, and closing strategies.

Reports can be sent to the screen or to the printer. They will run from one to two dozen screens of text or three to eight printed pages. Reports use your counterpart's name or initials and appropriately refer to "him" or "her." (You tell the program whether your counterpart is "Mr." or "Ms." when you ask for a report.)

I have numerous small complaints about the *Edge* programs. *Negotiation* prompts you to enter your counterpart's name. If you backspace past the first character you type, the program will crash, returning you to BASIC. And *Negotiation,* unlike the other three, will not allow you to update your assessment of a counterpart. You must reenter all your answers from scratch—especially annoying since *Negotiation* has the most comprehensive assessment (112 responses).

Management also had its quirks. Maybe I had a bad diskette, but ev-

erything involving the disk took longer. Generating a final report took a full 20 minutes–four times longer than the other programs. And after the report had printed, I tried to enter another assessment for a subordinate. Everything worked fine, except the screen was blank where the traits should have been listed. I could still agree or disagree, but with what? The same thing happened when I tried to update my self-assessment after printing a report.

Sales has no "next" option in the self-assessment. That's no problem, unless you are updating an existing assessment. You will have to go backward through the statements with the "previous" command. *Communication*, I am glad to say, had no glitches. It did, however, like all the others, contain occasional spelling and capitalization errors. Those were not too annoying in the text of final reports, but when you are rating a counterpart and have to agree or disagree that he is "derendable," your concentration will certainly be broken.

One real limitation of the assessments is that they only allow you to agree and disagree. This limitation might affect the validity of the final report. I would have preferred strongly agree, somewhat agree, no opinion, somewhat disagree, and strongly disagree. In assessing others, I often found that I did not know if they were "moody," or "organized," or "clever." I would have entered "no opinion" if given the choice.

In assessing myself, I had trouble with some statements like "I would rather 'win' than make another person happy." If I win, will that person be unaffected (though not "happy") or will he be broken? Or, "My success depends more on others and luck than on my own skill." As a manager, I'd say my success depends on my skill or ability to get work done through others. So the right answer must be....

Each of the *Edge* programs comes with a brief (13- to 17-page) booklet that describes the program generally and tells how to load and run it. Since the programs are menu-driven, instructions are almost unnecessary. Each program also has a screen or two of online instructions available at the main menu. What is missing is any guidance in the areas of communication, management, negotiation, or sales beyond what is printed in the strategy reports.

To the degree that those reports tell you to do specific things, you will be okay; unless your interaction calls for a change in style. But the reports do not teach communication, etc. They give you general advice, not specific examples or exercises to develop your skills. Available from Info Designs for $15.95 each are booklets on the fundamentals of management, negotiation, or sales. I thought those should have been included in the $79.95 purchase price, however.

The *Edge* programs come with a standard 90-day warranty. Backup disks cost $12.50 when you mail your warranty card, and $25 any time after that. Support for the program costs $25 per hour, though unless they intend to teach management over the phone, you should need no support to use this software.

The *Edge* series has been great fun and I intend to try some of its advice. Anticipating the recommendations in the strategy reports was fascinating. Even when I disagreed with some part of a report, it caused me to analyze my interactions much more closely than I normally would. *Edge* programs are fast to learn and easy to use. But they really only give you food for thought and are not professional development tools. And for $79.95, I wish they would fix the typos and bugs and throw in the booklets on fundamentals.

Info Designs, Inc., 445 Enterprise Court, Bloomfield Hills, MI 48013 (phone: 1-800-445-INFO).

–Richard Herring

KEYS TO RESPONSIBLE DRIVING
CBS Software, Inc.
Commodore 64
Disk; $79.95

For most people, getting a driver's license is an important event. Being able to drive can give you a new sense of independence, and for teenagers it is a modern "rite of passage"

marking the transition into adulthood. Simply passing your driving test, however, doesn't necessarily make you a responsible driver. Just think of how many times licensed drivers have cut you off to avoid missing their exit. What *does* make you a responsible driver is knowing how to drive legally, defensively, and safely.

Keys to Responsible Driving is a self-contained, self-paced program designed to provide you with the knowledge necessary not only to pass your driving test, but, more importantly, to help you become both a safe and responsible driver. Even if you already have your driver's license, chances are *Keys to Responsible Driving* could teach you a lot.

The program is divided into three sections, including a pretest to measure your knowledge when you start, chapters that can be studied in any order and at your own pace, and a posttest that you can take to see how much you've learned once you complete all nine chapters. Like the written tests given by the motor vehicle departments of most states, the program's pre- and posttests consist of several multiple choice and true/false questions covering everything from the colors of road signs to the effects of alcohol on driving ability.

When you select "execute a chapter" from the main menu, you are given a choice of nine chapters to study. They are:

1. Control Skills, Handling and Maneuvering
2. Signs, Markings and Regulations
3. Seeing and Reacting
4. City and Town Driving
5. Changing Lanes and Turning
6. Open Roads, Highways and Country Driving
7. On the Expressway
8. Unfit to Drive
9. Higher-Risk Driving

All chapters are presented in an interactive question and answer format and make excellent use of the C-64's graphic capabilities. Chapter one, for example, uses several pictures to illustrate the proper way to hold the steering wheel while turning, how to make two-point turns, and the various methods of parking. In chapter three, several hazardous traffic situations are depicted and you are asked about the best way to respond in each case.

Sound is also used in several of the chapters to help bring the material to life. For instance, in chapter three you are given a test of your reaction time. You are instructed to press any key when you hear the tone. When the tone sounds, a colored bar begins growing rapidly across the screen and stops once a key is pressed. By letting you see how long the bar gets in the time it takes you to respond to the sound, the program illustrates the delay between when you first notice a dangerous situation ahead of you on the road, and the time it takes you to react to it.

The program also comes with a thirty-page "Guide to Responsible Driving" which reiterates many of the points illustrated in the chapters. It then goes on to discuss a number of other issues related to responsible driving such as what to do at the scene of an accident, insuring that your car is in good condition, and state laws concerning traffic and safety violations. It includes a glossary of driving and automotive terms and is the perfect supplement to an excellent program.

CBS Software, One Fawcett Place, Greenwich, CT 06836 (phone: 203-622-2500). *—Bob Guerra*

BUSINESS LETTERS
BCI Software
Commodore 64
Disk; $4.99

Throw away your word processor! Or at least keep it safely in its sleeve the next time you need to grind out the typical letters that comprise most of a small business's outgoing mail. Over 50 types of business correspondence can be prepared with *Business Letters,* a stand-alone program with a fill-in-the-blank design so easy to use that even the boss may be able to handle it.

First you answer a series of prompts with your name, the company's name, address, phone number, and date. (Get it right the first time, for you won't get a chance to edit this

data.) Next you'll be asked if you want to include either or both optional insertions: an Attn: slot at the top of the body of the letter, and your job title (which appears below your name at the end of the letter). If you decide not to proceed, you may also back out and return to the main menu from this menu.

Otherwise the main menu shows up and offers nine general topics: letters of favour, inquiry, payment, sales, credit, complaint, collection, application, and miscellaneous. Punch a number and, after brief disk access, a submenu lists the specific letters in the chosen category. Most consist of five or six forms that vary in purpose or tone. Collection letters, for instance, include 13 different versions of "pay up." There is collection letter number one, then the one you'll send if the first is ignored, and three more —each a bit more insistent. (If you've ever bought anything on credit, this will sound familiar.) Then there's a so-called humorous letter that says, "Please send us the name of a lawyer in your town. We may have to sue you." And of course, the inevitable "have you forgotten?" (about the money you owe us) letter.

After choosing a letter from this menu, you wait a second while it loads. The letter is displayed, with blank lines embedded in the text. The first four blanks are for the name and address of the person you're writing. A prompt asks you to fill in the blanks, one by one, which is accomplished by typing the information at the bottom of the screen. A payment letter might read "I'm enclosing our cheque for $---e--- for the --f--." When prompted to fill in line e, you type the correct number. The name of the product goes in line f. Commas and colons may be included in an entry by beginning it with a quotation mark; entries may be up to 80 characters long.

The only snag with this process is that when long letters are displayed they automatically scroll down and cut off the top part of the text, so sometimes you can't see part of the letter in which blank e is mentioned. It's impossible to scroll up or page back to review it, so you won't know what to enter there until you've printed at least one of these letters.

You can also personalize the contents of a letter by deleting or typing over the existing text. Spaces may be inserted if you need to add more text, but only up to the current length of a line. (In other words, you can't use the "insert" key to push the last word in a line down to the next line.) You can't save a letter that has been modified in this manner. When all the blanks have been filled in, the program tosses your information into the dotted lines, displays the results, and asks if you need to change the text in any of the blanks. If you decide to do so, a list shows the contents of each blank and lets you choose which you want to revise or replace. (But you can't go back and display the entire letter to see your changes in context.)

If you have no more changes to make, the program prints your letter, which rolls out of the printer as a standard modified block letter: your name and address are centered atop the page, the date and closing (Yours very truly) are set on the right side of the page, and the name and address of the person to whom the letter is addressed is on the left. Text is single-spaced, with a blank line between paragraphs; each paragraph is indented five spaces. (None of these settings are adjustable.) Margins are also preset, so you have to align the paper before printing.

After the letter is printed, you can print the same letter with a new address, return to the main menu to select a new letter, or quit the program. The first option is handy when dealing with a small number of letters on the same subject, but there is no way to insert names and addresses from a database and send out personalized form letters in volume.

The program was apparently written in Canada or England, so American users will have to edit the British spelling of words such as cheque and favour. You can't add notes about enclosures or copies sent to other people at the bottom of a letter. And the "documentation," all two paragraphs, doesn't say a word about printer compatibility. With my TTX 1014 letter quality, the ? (which always precedes the program's prompt for information) was sometimes printed in my letter. And since there is no way to insert escape codes, you won't be able to underline or take advantage of your printer's particular features. I assume there will be no such problem with dot matrix printers, especially Commodore models—but you might want to be sure before buying the program.

And who would buy it? Anyone whose word processing time is devoted to generating a variety of form letters *one at a time*. Certainly you could sit down with a word processor and create your own form letters, placing blank lines in the appropriate spaces—but why bother when you can get 50 letters ready to go for a mere $5.00? (The letters are all sequential files, too, which means they can be loaded with most word processors, edited to meet specific needs, and then resaved for later use with the word processor. However, this eliminates the convenient fill-in-the-blank feature.)

BCI Software, P.O. BOX 730, Ringwood, NJ 07456 (phone: 201-835-7300). —*Shay Addams*

PROGRAMS WANTED

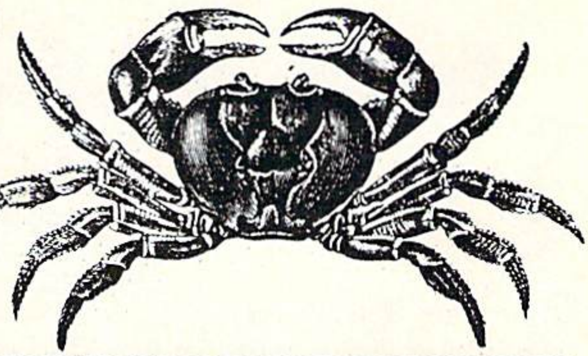
Don't be fooled by the fact that *Ahoy!* comes your way each month packed with the best games and utilities available in any Commodore magazine. We just don't have the mile-high inventory that you might imagine. But we're determined to continue bringing you the type of high quality programs you've come to expect from *Ahoy!* To that end, we've raised our rate of pay for accepted programs, and we've added staff to speed our response to your submissions. There's never been a better time to submit a program to *Ahoy!*

Send your program on disk or cassette (preferably disk), accompanied by an introductory article, a printout, and a self-addressed envelope of sufficient size with sufficient return postage affixed. (Submissions not accompanied by such an envelope will not be returned.)

Address submissions to *Ahoy!* Program Submissions Dept., Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001.

CRABFIGHT

For the C-64



By Buck Childress

We've all heard of the Gunfight at the O.K. Corral. Now there's the Crabfight in Commodore RAM. Two crabs, red and purple, are pitted against each other in a duel to reach 200 points.

Crabfight is a machine language game that must be entered using *Flankspeed* (see page 98). After saving *Crabfight*, reset or turn off/on your computer. Disk users must LOAD"CRABFIGHT",8,1 and tape users LOAD "CRABFIGHT",1,1. When the program has finished loading, "READY." will appear on the screen. At this point type NEW and hit RETURN. To begin execution of the program type SYS 49152 and hit RETURN.

Crabfight will prompt you for the level of play. One is the fastest and three is the slowest. Next you select one or two players. If two are playing, the game starts. If not, you're asked to choose the computer difficulty, one being the hardest and three the easiest. Use joystick port 2 when playing the computer. By the way, the computer crab is a very formidable opponent.

The game begins with a crab race. A clam is placed in the center of the playing area. Five warning beeps will sound, then the race is on. The first crab to reach the clam takes possession of it. Each player must try to protect an octopus and at the same time attempt to eliminate the opponent's octopus by shooting the clam at it. You start with two of them. When they or any others are lost, a new octopus will appear. You receive 20 points each time you hit an opposing octopus. You also get five points whenever you hit the wall behind your opponent.

To move your crab, push the joystick in the direction you want it to go. You can take possession of the clam by touching it with your crab. To shoot the clam, press the fire button and push the joystick left to shoot left or right to shoot right. If you're the red crab, pressing the fire button while pulling the joystick back will shoot the clam straight. For the purple crab, press the fire button and push the joystick forward. Whenever you press the fire button your crab will turn blue. Press it only when you intend to shoot the clam. Your crab will not move with the fire button depressed and a quick opponent, observing your blue crab, will score a lot of points.

Once you have the clam, you are given three seconds (in crab time, of course) to get rid of it. A timer will count down from three. If it reaches zero, you'll lose possession of the clam and your opponent gets ten penalty points. The clam is returned to the center of the playing area and a new crab race takes place.

Crabfight can be frozen in play by pressing RUN/STOP. A message will appear reminding you that pressing the N key will take you back to the opening menu for a new game, while pressing the C key will continue the game in progress. The message also appears at the end of the game. At this time, pressing the C key will start another game with the same levels and players as the previous game. Pressing the N key takes you back to the opening menu. □

SEE PROGRAM LISTING ON PAGE 115

Lyco Computer Marketing & Consultants

"WE MAKE YOUR COMPUTER FUN TO USE!"

NO LABEL DISKETTES

NL 5¼" SSDD...10.99 (Box 10)
NL 5¼" DSDD...15.99 (Box 10)
*Free Diskette Writer Pen!
*Free Storage Case!

DUST COVERS

Atari

520ST....11.95
130XE....6.99
800XL....6.99
1050....6.99
1025....7.99

Commodore

C128....7.99
1571/1541....6.99
1902....10.95
1702....8.99
C64/Vic20....6.99

Panasonic

1090/1091....8.99
1092....8.99
1093....9.99

Star Micronics

SG/SD10....8.99
SG/SD15....9.99
SR10....9.99
SR15....9.99

Okidata

82/92....8.99
83/93....9.99
193....9.99

PRINTING PAPER

3000 SHEETS FANFOLD....$42.75
1000 SHEETS FANFOLD....$19.75
1000 SHEET LETTER....$21.95
200 SHEETS LETTER....$8.99
150 RAG STATIONARY....$10.99
MAILING LABELS (1in.)....$9.95

WICO Joysticks

15-9714 Bat Handle....16.75
50-2030 Boss....11.99
50-2002 Super 3-Way....19.99

COMMODORE

C-128 NEW....CALL
1571 Drive....CALL
1572 Drive....CALL
1902 Monitor....CALL
1670 Modem....CALL
C-64 Computer....CALL
1541 Drive....189
MPS801 Printer....LOW
1702 Monitor....199
Simon's Basic....24.75
Assembler 64....34.75
Super Expander....22.75
Logo 64....49.75
Pilot 64....38.75
Easy Calc....34.75
Easy Script....38.75

COMMODORE SOFT-WARE

MICROPROSE (C-64)

Kennedy Approach....21.75
Crusade in Europe....24.75
Decision in Desert....24.75
Solo Flight....20.75
Nato Commander....20.75
Spitfire Ace....18.75
F-15 Strike Eagle....20.75
Hellcat Ace....18.75
Acrojet....21.75
Silent Service....21.75

CARDCO

Digitizer Camera....189.95
32K Printer Buffer....59.95
Numeric Keypad....34.95
CB/5 5-slot Board(64)....54.00
CB/2 2-slot Board(64)....25.00
S'More Basic Rom....49.95
Write Now-64....35.00
Mail Now-64....29.00
Spell Now-64....29.00
File Now-64....29.00
Paint Now-64....29.00
Calc Now-64....29.00
Tax Survival....29.00
Super Printer Utility....27.95
Write Now-Vic-20....29.95

BRODERBUND

The Print Shop....28.75
Graphics Library....18.75
Graphics Library II....19.75
Graphics Library III....19.95
Karateka....19.75
Castles Dr. Creep....19.75
Bank St. Writer....32.75
Loderunner....20.75
Mask of the Sun....24.75
Spelunker....19.75
Serpent's Star....24.75
Whistler's Brother....18.75
Raid Bungeling Bay....18.75

COMMODORE

MPS1000 Printer....259
C1350 Mouse....42
C1700 128K RAM....145
C1750 512K RAM....269
JANE....35
Perfect Writer....49
Perfect Calc....49
Perfect Filer....49

SPINNAKER (C-64ROM)

Cosmic Life ROM....19.75
Jukebox....19.75
Alphabet Zoo....17.75
All in Color Caves....19.75
Up for Grabs....19.75
Delta Drawing....19.75
Kids on Keys....16.75
Kindercomp....14.75
Facemaker....17.75
Fraction Fever....17.75

SSI (C-64)

Colonial Conquest....24.75
Wings of War....24.75
Computer Ambush....34.75
Field of Fire....24.75
Fighter Command....36.75
Kampfgruppe....36.75
Mech Brigade....36.75
Market Garden....29.75
Six Gun Shootout....24.75
Computer Baseball....24.75
Computer Quarterback....24.75
Imperium Galactum....24.75
Phantasie....24.75
Cartels & Cutthroats....24.75
50 Mission Crush....24.75
Questron....24.75

INNOVATIVE CONCEPTS

Flip-n-File 10....3.50
Flip-n-File 15....8.25
Flip-n-File 25 Lock....17.95
Flip-n-File 50....17.25
Flip-n-File 50 Lock....22.95
Flip-n-File Rom....17.25

SCARBOROUGH (C-64)

Build A Book....24.75
Improved Mastertype....23.75
NET WORTH....48.75
Mastertype Filer....22.75
Boston 64 Diet....27.75

TRONIX

S.A.M. - Atari....38.50
S.A.M. - C-64....38.50

PERSONAL PERIPHERALS

Super Sketch 64....32.75
Printer Utility....18.75

BATTERIES INCLUDED

Paper Clip....59.95
Spell Pak....34.95
Consultant....59.95
Paper Clip w/Spell Pak....75.95
Home Pak....34.95
Bus Card....129.95
80 Column Board....109.95

EPYX

(C-64)

Fast Load....26.75
Breakdance....23.75
Greatest Baseball....24.75
Summer Games....26.75

SUB LOGIC (C-64)

Flight Simulator II....32.75
Night Mission Pinball....20.75

CONTINENTAL

(C-64)

Home Accountant....44.75
1984 Tax Advantage....35.75
1985 C-64 Book of Software....16.95

QR & D

Copy Q....27.95
GPC Printer Interface....65.00

EASTERN HOUSE

Rabbit C-64....19.95
Rabbit VIC-20....19.95
MAE C-64....27.95
Telstar 64....19.95
M.L. Monitor 64....18.95

KOALA

(C-64)

Koala Pad....59.95

COMPUTER CARE

NORTRONICS DISK DRIVE CLEANER
with software

REG. 49.95 NOW 19.95

BUY LYCO AND ENJOY

★ THE LOWEST PRICES ★ TOLL FREE ORDER LINE ★
★ Free shipping on prepaid cash orders in U.S. ★ All Merchandise Factory Fresh ★
★ 24 hrs. shipping on in-stock product ★ Access to our Multi Million $ inventory ★
★ No deposit on UPS C.O.D. orders ★ Orders outside PA save state sales tax ★
★ Air freight service available ★ Full Manufacturer's Warrranty apply! ★ Full accessory line in stock ★
★ Purchase Orders Accepted from educational institutions! ★ We check for stolen credit cards! ★
★ We ship to our servicemen overseas! ★ You'll love our Courteous Sales Staff! ★

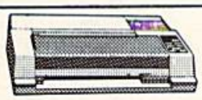
AMERICA'S MAIL ORDER HEADQUARTERS

LYCO COMPUTER

WORLD'S LEADER IN SALES & SERVICE

TO ORDER
CALL TOLL FREE
800-233-8760
In PA 1 717-327-1824
Lyco Computer
P.O. Box 5088
Jersey Shore, PA 17740

Lyco Computer Marketing & Consultants



SAVE ON THESE IN STOCK PRINTERS



1091...........$233

SG-10.........$208

AXIOM SEIKOSHA
- GP550AT (Atari) ... 222
- GP550CD (C-64 ... 222
- GP700AT (Atari) ... 439
- GP700AP (Apple) ... 439
- ELITE5CD (C-64) ... 229

C. ITOH
- ProWriter 8510Sp+ ... 349
- 1550Sp+ ... 489
- StarWriter ... 769
- PrintMaster ... 929

TOSHIBA
- P1340 ... 469
- P351+ ... 1149
- P341P ... 969
- P341S ... 999
- 351-Sheet Feeder ... 529

CARDCO
- 32K BUFFER (C-64) ... 59

CORONA
- LP300 Laser Printer ... 2686
- 200361 Toner Cartridge ... 89

EPSON
- FX85 (New) ... 333
- LX80 ... 212
- FX185 (New) ... 464
- LX90 (New) ... 226
- SQ2000 (New) ... 1555
- JX80 ... 467
- Homewriter 10 ... 193
- CR-20-Atari ... 153
- CR-220-C-64 ... 153
- DX-10 (New) ... 207
- DX-20 (New) ... 297
- HS-80 (New) ... 288
- LQ1500P ... 975
- LQ1500S ... 1039
- RX-100 ... 356
- FX-100+ ... CALL

CITIZEN
- MSP-10 ... 269
- MSP-15 ... 358
- MSP-20 ... 337
- MSP-25 ... 495
- Sheetfdr (10/20) ... 189
- Sheetfdr (15/25) ... 199

OKIDATA
- Okimate 10 ... 179
- Okimate 20 ... CALL
- 182 ... 214
- 192 ... 348
- 193 ... 563
- 92P ... 349
- 93P ... 565
- 84P ... 645
- 92 Imagewriter ... 349

*IBM versions also

BROTHER
- HR-15XL-P ... 359
- HR-15XL-S ... 359
- HR-35P ... 839
- HR-35S ... 839
- 2024L-P ... 949
- M1009-P ... 189

JUKI
- Juki 6100 ... 347
- RS232 Serial Board ... 55
- 6100 Tractor ... 119
- 6100 Sheet Feeder ... 209
- Juki 6300 ... 757

LEGEND
- 880 ... 188
- 1080 ... 222
- 1380 ... 262
- 1385 ... 296
- LEGEND 808 ... 159

DIGITAL DEVICES
- 16K BUFFER ... 75
- 32K BUFFER ... 89
- 64K BUFFER ... 125

DIABLO
- D25 ... 549
- 630 API ... 1599
- 630 ECS ... 1759
- D 80 1F ... 2395
- P 32 CQ1 ... 699
- P 38 ... 1749
- C 150 ... 999
- DX-35 (NEW) ... CALL
- AP-80 ... CALL

PANASONIC
- ***1091 ... 233***
- 3131 (NEW) ... 269
- 1092 ... 373
- 1093 ... 426
- 3151 Letter ... 426
- 4K Buffer ... 65

SILVER REED
- EXP400 ... 249
- EXP500 ... 295
- EXP550 ... 399
- EXP770 ... 749

STAR MICRONICS
- SG-10 ... **208**
- SG-15 ... 373
- SD-10 ... 336
- SD-15 ... 442
- SR-10 ... 483
- SR-15 ... 583
- SB 10 ... 595
- Power Tvpe ... 303
- SG-10 C-64 (NEW) ... CALL

MONITORS

TAXAN
- 115 12" Greeen Composite ... CALL
- 116 12" Amber Composite ... CALL
- 121 12" Green TTL ... 135
- 122 12" Amber TTL ... 145
- 220 14" Color Composite ... 259
- 410 12" RGB Hi Res IBM ... 329
- 420 12" RGB Super Hi IBM ... 409
- 440 12" RGB Ultra Hi Res ... 555
- Tilt Stand ... 35

ZENITH
- ZVM 122A Amber ... 75
- ZVM 123G Green ... 75
- ZVM 124 Amber IBM ... 129
- ZVM 131 Color ... 275
- ZVM 133 RGB ... 389
- ZVM 135 Composite ... 449
- ZVM 136 Hi Res Color ... 589
- ZVM 1220 ... 95
- ZVM 1230 ... 95
- ZVM 1240 ... 149

TEKNIKA
- MJ-10 Composite ... 179
- MJ-22 RGB ... 255

AMDEK
- 300 Green ... 118
- 300 Amber ... 128
- 310 Amber IBM ... 155
- Color 300 Audio ... 234
- Color 500 Composite ... 369
- Color 600 ... 397
- Color 700 ... 495
- Color 710 ... 569

PANASONIC
- DT1300D 13" RGB/Composite ... 247
- DTM140 14" RGB/Composite ... 329
- DTH103 10" RGB Hi Res ... 395
- DTS101 10" Composite ... 175
- DT1000G 10" RGB ... 166
- TX12H3P 12" Color ... 419
- TR120M1PA 12" Green ... 109
- TR120MBPA 12" Amber ... 109
- TR122M9P 12" Green IBM ... 148
- TR122MYP 12" Amber IBM ... 148

SAKATA
- SG 1000 12" Green ... 99
- SA 1000 12" Amber ... 109
- SG 1500 12" Green TTL ... 119
- SA 1500 12" Amber TTL ... 129
- SC 100 13" Color Comp ... 209
- SC 200 13" RGB ... 389
- STS1 Tilt Stand ... 29

X-TRON
- Comcolor I Composite Green ... 177

NEC
- JB-1260 Green ... 95
- JB-1201 Green ... 135
- JC 1215 Color ... 235
- JC 1216 RGB ... 375
- JC 1460 Color ... 265

PRINCETON GRAPHICS
- MAX-12 Amber ... 185
- -12 RGB ... 465
- SR-12 RGB ... 595

INTERFACING

DIGITAL DEVICES
- U-Print C (C-64) ... 49

ORANGE MICRO
- Grappler CD (C-64) ... 79

QRD
- GPC (C-64) ... 59

CARDCO
- G-Wiz (C-64) ... 54
- C/? + G (C-64) ... 45
- C/?PS (C-64) ... 49
- C/?B (C-64) ... 39

TYMAC
- Connection (C-64) ... 55

MODEMS

HAYES
- Smartmodem 300 ... 133
- Smartmodem 1200 ... 377
- Smartmodem 1200B ... 347
- Smartmodem 2400 ... 598
- Mibromodem IIE ... 135

MICROBITS
- 1064Modem ... 57

TELE LEARNING
- Total Telecommunications (C-64) ... 29.95
- AP-250 (300 Baud Apple) ... 69.95
- IB-250 (300 Baud IBM) ... 69.95

US ROBOTICS
- Password 1200M ... 229
- Password 1200F ... 229
- Password 300M ... 139
- Password 300F ... 139
- Autodial 212A ... 359
- PCM5 ... 319
- PCM64 ... 519
- PCM256 ... 769
- S-100 ... 379
- Courier ... 469
- Microlink ... 469
- Telpac Ms-DOS ... 79

NOVATION
- IBM 300/1200 MS-DOS ext ... 319
- IBM 300/1200 CPM-86 ext ... 319
- IBM 300/1200/2400 ext ... 529
- IBM 300/1200/2400 MS-DOS ... 579
- IBM MS-DOS int ... 325
- IBM CPM-86 int ... 325
- Macmodem 300/1200 ... 315
- Upgrade Apple Cat II ... 225
- Cat 300 Acoustic ... 139
- J-Cat RS232 ... 89

ANCHOR
- Volksmodem ... 55
- Volksmodem 12 ... 186

DRIVES

COMTEL
- Enhancer 2000 (c-64) ... 179

INDUS
- ***GT ATARI ... 215***
- GT COMMODORE ... 235

MSD
- SD1 Drive (C-64) ... 229
- SD2 Drive (C-64) ... 469

DISKETTES

DENNISON
- ELEPHANT 5¼" SSSD ... 11.99
- ELEPHANT 5¼" SSDD ... 12.99
- ELEPHANT 5¼" DSDD ... 14.99
- PREMIUM 5¼" SSDD ... 13.99
- PREMIUM 5¼" DSDD ... 15.99

SUNKYONG
- SKC 5¼" SSDD ... 11.99
- SKC 5¼" DSDD ... 13.99

MAXELL
- 5¼" MD1 ... 13.99
- 5¼" MD2 ... 19.99

VERBATIM
- 5¼" SSDD ... 13.99
- 5¼" DSDD ... 19.99

BONUS
- 5¼" SSDD ... 9.99
- 5¼" DSDD ... 12.99

TOLL FREE 1-800-233-8760

MasterCard

TO ORDER

CALL TOLL FREE

800-233-8760

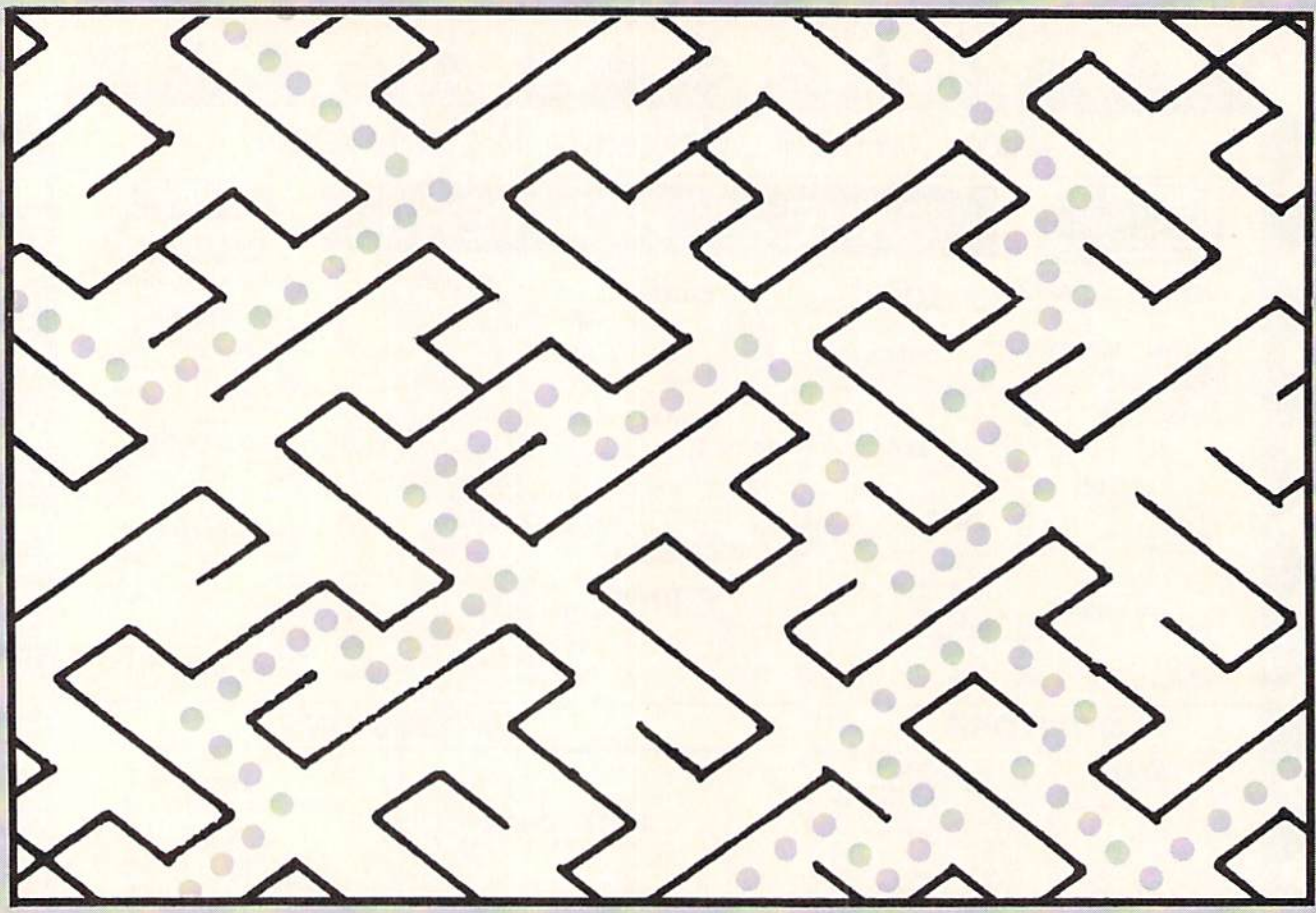
Customer Service 1-717-327-1825

VISA

or send order to
Lyco Computer
P.O. Box 5088
Jersey Shore, PA
17740

RISK FREE POLICY

In-stock items shipped within 24 hours of order. No deposit on C.O.D. orders. Free shipping on prepaid cash orders within the continental U.S. Volume discounts available. PA residents add sales tax. APO, FPO, and international orders add $5.00 plus 3% for priority mail service. Advertised prices show 4% discount for cash, add 4% for MasterCard or Visa. Personal checks require 4 weeks' clearance before shipping. Ask about UPS Blue and Red label shipping. All merchandise carried under manufacturer's warranty. Free catalog with order. All items subject to change without notice.



HEAD TO HEAD

Two-Player Maze Action for the C-64

By Doug Smoak

My first computer was a VIC 20 and I enjoyed playing and writing games for it. But one shortcoming of those games was that they were usually one person against the computer. You could have multiple players, but they were just taking turns playing against the computer. This was primarily because the VIC has only one joystick port.

When I got my C-64, I was so happy with its two joystick ports that I decided to write a game where two people could compete "head to head." This game is the result of that effort. It is the competition side by side on the screen that makes this game so much fun to play.

The format of the game is not that unusual: a maze filled with dots that you must guide your players through to gather points. The idea is to gather the dots before your opponent can. Since you play in identical mazes that are side by side on the screen, you don't really know where your opponent is until you see the dots in *your* maze disappearing.

The game has several options that may help to even out two players of different ages or skills. You can choose to play a number of rounds and have the score accumulate from each round until the game is over. You can choose to have a bonus (a large x) placed in the maze that will be worth 500 points (50 dots) and you can replay a particular maze over and over by answering the MAZE NUMBER? prompt with the same number over and over. If you just hit RETURN you will get a randomly chosen maze.

After the options have been chosen the computer will draw the two mazes on the screen and fill them with dots. When the mazes are finished a "tune" will start playing and a 10-second countdown will begin before you can start. This countdown gives you a chance to plan your strategy and intimidate your opponent with verbal abuse. When the game is over the screen will show scores and announce the winner.

While this may seem like a friendly and harmless little game, I will not be responsible for broken friendships, marriages, or bones when you play *Head to Head.*

The listing has some comments in it to help you in debugging your program if you type it in and have problems. □ *SEE PROGRAM LISTING ON PAGE 119*

MEMORY MAGIC

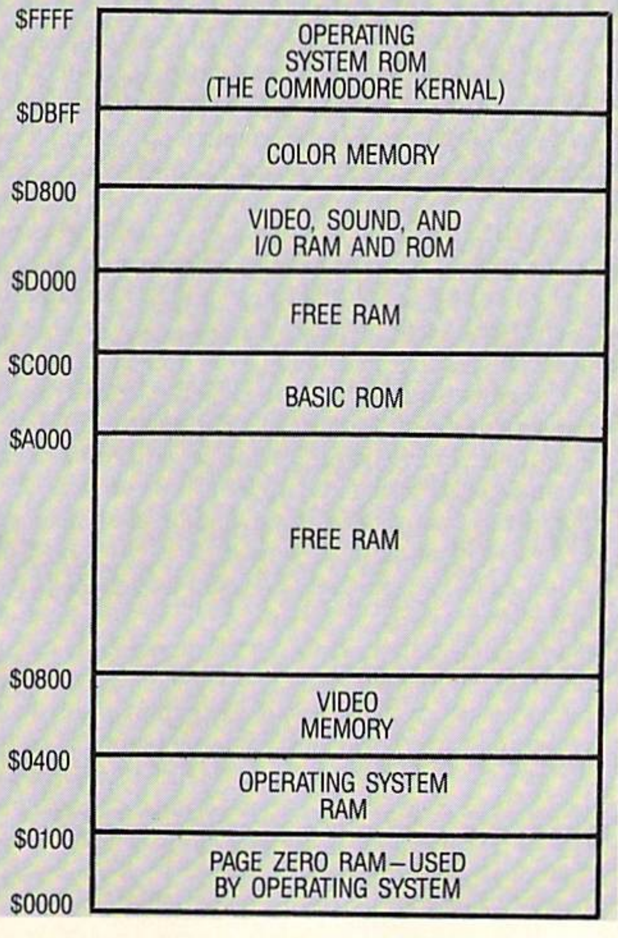
By Mark Andrews

The engineers who designed the Commodore 64 accomplished quite a feat: they stuffed 88 kilobytes of memory in a 64K machine. So, from a memory-management point of view, the C-64 is a rare breed of computer. Most 64K computers have only 48K or so of addressable RAM, plus around 16K of ROM, for a total of 64K. But the C-64 has a full 64K of user-addressable RAM, plus 24K of built-in ROM, for a total of 88K of memory.

This 88K is controlled by a pair of memory locations that occupy addresses $0000 and $0001 (that's just plain 0 and 1 in decimal notation, if you're not familiar with hexadecimal). With the help of these two memory registers, a skilled programmer can wield a tremendous amount of control over a 64. By simply switching certain bits on and off in this pair of registers, you can actually determine which portions of your computer's memory will be used as RAM, and which blocks of memory will be used as ROM. And, as we shall see before this column is finished, that's a tremendous amount of power for a computer programmer to have.

If you really wanted to, you could use memory registers $0000 and $0001 to switch off every byte of ROM in your C-64. Then, if you had the necessary programming skill, you could turn your computer into a totally customized machine, with a homemade operating system and a set of input/output drivers of your own design. We're not going to get nearly that ambitious, but we will be dealing with one pretty fancy trick that involves switching back and forth between RAM and ROM. In this month's edition of *Commodore Roots,* you'll learn how to copy your computer's built-in character set from ROM into RAM. Then, next month, you'll learn how to modify this new character set and turn it into a set of custom-designed text or graphics characters.

FIGURE 1
A SIMPLIFIED MEMORY MAP OF THE COMMODORE 64 (NOT TO SCALE)



YOUR COMPUTER'S MEMORY MAP

Before we start doing all this, let's take a look at a memory map of the C-64. Figure 1 is a simplified map that illustrates your C-64's default memory configuration—what the computer's memory map looks like when the power is first turned on.

Addresses $0000 through $00FF. The block of memory that extends from $0000 through $00FF—often referred to as Page Zero—is so desirable that the designers of the C-64 claimed most of it for themselves. The operating system, the BASIC interpreter, and other essential ingredients of your computer system take up most of the available space on Page Zero. Consequently, there's very little room left on Page Zero for use in other kinds of programs.

This shortage of space on Page Zero can create tough problems for the assembly language programmer. It restricts the use of Page-Zero addressing, which can make programs run faster, and it also makes it difficult to use indirect indexed addressing, which will not work at all unless space on Page Zero is available. To write the best possible assembly language programs, therefore, a programmer just has to find at least a few free memory addresses on Page Zero. And, as you can see by looking at the following list, that can be quite a chore.

This column was adapted especially for Ahoy! *from Mark Andrews' book,* Commodore 64/128 Assembly Language Programming, *published in 1985 by Howard W. Sams & Co. Inc., Indianapolis, IN, and printed with the permission of Sams.*

USEABLE PAGE-ZERO ADDRESSES

Memory Locations	Descriptions of Registers' Functions
$00-$01	Special 6510 processor I/O addresses
$02	Not used
$03-$FA	Registers used by BASIC and the C-64 operating system
$FB-$FE	Bytes left free for user-written programs
$FF	Used by BASIC interpreter

As you can see, there are only four bytes on Page Zero —$FB, $FC, $FD, and $FE—that are always free for use in user-written programs. But there are quite a few Zero-Page addresses that can be used safely in certain situations. For example, many of the addresses on Page Zero are reserved for use by the Commodore 64 BASIC interpreter, and a number of other Page-Zero addresses are used only by the floating-point arithmetic routines that are built into the 64. So, in assembly language programs that are not called from BASIC and do not make use of BASIC's floating-point routines, there are actually quite a few Page-Zero addresses that can be used. A complete list of these addresses can be found on the memory map starting on page 310 of the *Commodore 64 Programmer's Reference Guide.*

Addresses $0100 through $01FF (The Stack). Page One of the 64's memory—that is, the segment of memory that extends from $0100 through $01FF—is a special bank of memory called the stack. This is the section that the C-64's 6510 microprocessor uses as a temporary storage area during the processing of machine language programs. The stack is heavily used by the C-64 operating system, and is also available for use as a temporary storage area by user-written programs. Before you can use Page One safely, however, it is essential to have an understanding of how the stack works in assembly language programs. Under ordinary conditions, the stack cannot be used at all for permanent data storage. And, unless you have a thorough understanding of stack operations, it's best to stay completely away from Page One.

Addresses $0200 through $03FF. Most of the memory space that extends from $0200 through $03FF is reserved for use by the Commodore 64 operating system. However, there is one small block of memory in this area that is usually free for use in user-written programs. This free block, 88 bytes long, extends from memory address $0237 through memory address $02FF. This segment is memory is too short to do much programming in, but it can be a useful spot for storing data tables.

Addresses $0400 through $07FF. When you turn on your Commodore 64, the segment of memory that extends from $0400 through $07FF is the block that your computer uses as a memory map—that is, for the storage of data which it uses to generate its screen display. When the C-64 is in its high-resolution mode, however, this area is not nearly large enough to hold the amount of data that is needed for a screen map; it has only 1,000 bytes of storage space, while a high-resolution map requires 8,000 bytes. A high-resolution screen map can be stored in almost any block of free RAM that's big enough, and it's the programmer's responsibility to find one. Since the $0400-$07FF block of memory is not large enough to be used for high-resolution screen-mapping, it is generally used for another purpose in high-resolution programs: specifically, to determine what colors will be used in the C-64's high-resolution screen display.

Addresses $0800 through $9FFF. The 38K block of memory that extends from $0800 through $9FFF is free RAM, specifically set aside for use by user-written or commercial applications programs. When you write a program—or buy one—this is the area of memory in which it will usually be stored.

At first glance, this looks like quite a big hunk of memory. Unfortunately, however, the closer you look at it, the smaller it gets. When you write an assembly language program, your assembler, editor, and machine language monitor usually consume a large portion of this section of memory. And when you write a program that requires the use of both assembly language and BASIC, you can cramp your style even more, since you then have to take special precautions to keep your BASIC and machine language programs from running into each other. Some tips on how to separate the BASIC and machine language por-

tions of a program will be provided later in this series of columns.

Addresses $A000 through $BFFF. When you turn on your 64, memory addresses $A000 through $BFFF are usually occupied by your computer's BASIC interpreter. If you don't need BASIC in a program, however, you can switch this block of memory from ROM to RAM, and can thus add 8K of user-accessible RAM to your computer's memory. Details on how to do this will also be provided later in this column.

Addresses $C000 through $CFFF. In memory locations $C000 through $CFFF, there is another 4K of RAM that's theoretically free for use in user-written programs. You sometimes have to be careful with this block of memory, though, since several popular C-64 assemblers – including the *Commodore 64* assembler, the *Merlin 64*, and the *Panther C-64* – all make use of it in one way or another. If you know how to stay out of your assembler's way, however, this is a useable segment of your computer's memory.

Addresses $D000 through $D800. Memory registers $D000 through $DFFF serve double-duty functions in the Commodore 64. With the help of some fairly sophisticated bank-switching techniques, these addresses are used as both RAM and ROM by the C-64's operating system. When they are used as RAM, their primary function is to help control input/output devices. When they are used as ROM, they hold the data which the C-64 uses to print characters on its video monitor. The techniques used to switch this block of addresses back and forth between RAM and ROM will be explained below.

Addresses $D800 through $DBFF. When the Commodore 64 is in its text mode, memory addresses $D800 through $DBFF are used for the storage of color data: the data that determines the colors of characters displayed on the screen. When the C-64 is in its high-resolution mode, this area of RAM is not used for color data and is free for other uses.

Addresses $DC00 through $FFFF. This block of memory is occupied by the Commodore 64 Kernal, a collection of machine language input/output routines that are extensively used by the C-64 operating system and are also available for use by user-written programs. If you had a need to, you could switch the C-64's Kernal ROM out of this area and use it as free RAM – but if you did that, you'd have to write your own operating system. A detailed explanation of the C-64 Kernal and how to use it can be found in Chapter 5 of the *Commodore 64 Programmer's Reference Guide.*

ANOTHER LOOK AT $0000 AND $0001

As pointed out earlier, programs that switch back and forth between RAM and ROM often make use of memory registers $0000 and $0001. In literature about the Commodore 64, memory register $0000 is often referred to as the computer's data-direction register, or as Register D6510. The D6510 register, as its name implies, is used to control the direction of the flow of data into and out of specified blocks of memory. It is also used to control the direction of data flow to and from the Commodore 64 datacassette recorder.

Memory Location $0001 is the Commodore 64's input/output port, or control port. It is often labeled R6510 in C-64 assembly language programs. The chief function of the R6510 register is to determine which blocks of memory will be used as RAM, and which as ROM, during the operation of a program.

The R6510 register and the D6510 have eight bits each, but only the lower six bits in each register – that is, Bits 0 through 5 – are significant. And there is a direct one-to-one correspondence between each of these bits in each register. The six significant bits of the R6510 register are used to control the types of I/O data that flow to and from the C-64. And each significant bit of the data direction register controls the direction of the flow of this data. If a bit of the D6510 register is set to 1, then data which flows through the corresponding bit of the R6510 register is output data. If a bit of the D6510 register is cleared to 0, then the data which flows through the corresponding bit of the R6510 is input data. On the facing page is a table listing the six significant bits of the D6510 and R6510 registers, along with their functions.

THE MAGIC BIT

The table on page 76 shows quite clearly what Bits 0, 1, 3, 4, and 5 of the R6510 and D6510 registers do. But Bit

THE R6510 AND D6510 REGISTERS

Bit	Name	Setting at Power-Up	Function
0	LORAM	1 (Output)	On: $A000-$BFFF is BASIC ROM Off: $A000-$BFFF is RAM
1	HIRAM	1 (Output)	On: $E000-$FFFF is Kernal ROM Off: $E000-$FFFF is RAM
2	CHAREN	1 (Output)	On: $D000-$DFFF is I/O ROM Off: $D000-$DFFF is character ROM
3		1 (Output)	On: Write to cassette line Off: Read from cassette line
4		0 (Input)	On: Cassette switch pressed Off: Cassette switch not pressed
5		1 (Output)	On: Cassette motor on Off: Cassette motor off

2 – the CHAREN bit – is worthy of special note. It's the "magic bit" in the D6510 and R6510 registers; it determines whether memory addresses $D000 through $DFFF will be used as RAM registers by the C-64's operating system, or as character-generator ROM. When Bit 2 of the R6510 register is set, the D000-$DFFF block of memory is used as RAM, primarily by the portion of the operating system that controls the operation of I/O devices. When the CHAREN bit is clear, all RAM stored in the $D000-$DFFF area becomes temporarily inaccessible, and 4K of character-generator ROM – better known to most Commodore 64 users as the C-64 character set – is switched in. In memory management jargon, that kind of electronic hocus-pocus is called bank-switching. Here's how it works:

The Commodore 64's screen graphics are produced by a sophisticated microprocessor called a Video Interface Chip, or VIC-II. To generate the characters that it displays on the C-64's screen, this chip uses a 4K character set that is permanently stored in RAM addresses $D000 through $DFFF. However, the VIC-II chip does not need access to this character data all the time. Under ordinary conditions, the VIC chip has to refer to its character-generator data only 60 times each second, during a split-second screen blackout that is often referred to as a video refresh cycle. The rest of the time, the VIC chip is busy with other matters, and does not require access to its character data. Most of the time, therefore, the Commodore 64 keeps the character ROM on the $D000-$DFFF block of memory bank-switched out, and keeps a 4K block of RAM bank-switched in. Only during the video refresh cycle does the $D000-$DFFF memory block become character-generator ROM. This switching process takes place so rapidly, and so transparently, that it is hardly ever noticeable to the average Commodore user – or to the average Commodore programmer. As far as most programmers are concerned, the memory that extends from $D000 to $DFFF is nothing but RAM. But actually, as we have seen, it is a double-purpose block of memory that is sometimes RAM and sometimes ROM.

This bank-switching technique is an ingenious method for expanding the nominal memory of a computer. However, it can occasionally cause problems. Sometimes a programmer would like to have access to the C-64's built-in character set for a longer period of time than the brief video refresh cycle provides. For example, sometimes a programmer might want to copy the computer's character set from ROM into RAM, so that it can be modified and then used in a user-written program. And this is a situation in which Bit 2 of the R6510 chip can come in quite handy. To prevent the C-64's character data from being bank-switched out while it is being copied, all a programmer has to do is turn off the CHAREN bit while the data-duplication process is taking place, and then turn it on again. While the CHAREN bit is off, the character-generator ROMI in the $D000-$DFFF block of memory can be copied into RAM. Then, when the copying operation is finished, the CHAREN bit can be turned back on.

The listing on page 104, called MOVECHRS, is an assembly language program that turns the CHAREN bit off, copies the C-64 character set from ROM into RAM, and then turns the CHAREN bit on again. In next month's column, we'll see how this duplicate character set can be combined with high-resolution graphics and incorporated into some user-written programs. ☐

SEE PROGRAM LISTING ON PAGE 104

CADET'S COLUMN

Building Up to Structured Programming By Cheryl Peterson

A few months ago, I talked about programming and presented a mortgage program that was thrown together in a rather slipshod manner. The point was to show that a program doesn't have to be a work of art to work. Today, we'll examine two different versions of the same program, *Compuphobia.* The first version is an unstructured but functional quiz program. The second version is the perfected and organized version. (Well, since perfection is relative, it's relatively perfected.) You'll probably notice right away that the second is shorter. This is one objective of structured programming: to make prudent use of computer memory by shortening a program. We'll take a close look at how this is possible.

Another reason for structuring is to make the resulting program easier to modify for alternative purposes. We'll touch on that a bit, too.

The third point of our exercise is to have a little fun. Our program supposedly measures an individual's level of compuphobia. After you've typed it in, you can try it on your "phobic phriends."

But first, let's pretend that you're going to design a program that asks a series of questions and analyzes the answers to come up with a pseudo-scientific result. We want to devise questions whose answers give some indication of how well a person will interact with a computer. So the first step is to create a multiple choice test, similar to the ones given by schools. Then you must come up with a way to analyze the answers and give an evaluation of the total.

I devised a ten-question quiz with one bonus question. Using a rating system that weighs answers on a 0-3 scale, I came up with possible scores between 0 and 30. Since I think most people's reactions to computers fall into one of four categories, my test gives four classifications depending on the quiz results.

Throughout the remainder of this column we'll be referring to the two program listings found under the heading *Compuphobia–Slipshod Version* (page 100) and *Structured Version* (page 101).

PROGRAMMING

Once you've got the questions, the fun starts. Programming! PRINT statements are used to display the questions on the screen. To read the answers, we need to accept input from the keyboard. There are two ways to do this: GET or INPUT statements. I chose to use INPUT because it waits for a carriage return before accepting input. This allows the person taking the test to change his mind. A short bit about GET statements later when we examine the structured version of the program. For now, let's look at the messy one.

```
130 PRINT"USING MY MICROWAVE'S AUTOMATIC
TIMER AND PRE-PROGRAMMED"
131 PRINT"COOKING CYCLES BOTHERS ME"
132 PRINT"1) NONE"
133 PRINT"2) A LITTLE"
134 PRINT"3) SOME"
135 INPUT"4) VERY MUCH";A
```

Now that we have the answer to the first question (line 135), we need to evaluate the answer and quantify it. We could assign a different variable to each question and then figure it all out at the end, but I wanted to keep variables to a minimum. So each answer is assigned to A, evaluated, and then added to the total score T.

```
136 IF A=1THENT=0
137 IF A=2THENT=1
138 IF A=3THENT=2
139 IF A=4THENT=3
```

Using the IF/THEN statements, let us compare the answer to a particular value and then reassign a value to the score. In subsequent answer evaluations, we take the new value and add it to the total we're accumulating. When the test is complete, we'll have a total that reflects all the choices made.

Since we need to be sure there's no cheating, we make sure the answer is between 1 and 4. A PRINT statement is used to help illuminate the problem when an incorrect number is used.

```
148 IF A>0ANDA<5THEN GOTO 230
149 PRINT"TRY USING A NUMBER, 1-4":GOTO
130
```

Actually, while this appears to be the correct way to go about doing it, it isn't. This is just a ruse to show you an example of sloppy programming. While it checks for the right range, it doesn't evaluate for whole numbers. This means that in this version decimal values will be accepted: 2.195 or 4.9, for instance. An alternative would be

```
148 IF A=1 OR A=2 OR A=3 OR A=4 THEN GOT
O 230
```

These sequences are repeated until all the questions have been answered, except the last. Note the subtle changes in the values in lines 136-139, 236-239, and 336-339. These change since the reverse order of evaluation is needed for some of the questions.

The bonus question requires a different method of evaluation, because it is the only one that subtracts from the accumulated total.

The remainder of the program compares the total score with a base figure and then prints the appropriate evaluation.

THE STRUCTURED VERSION

The structured version is organized differently. The organization is still fairly loose, but repeating sections have been moved to subroutines at the end of the program. In BASIC, these are called "subroutines." In truly structured programming, they are usually called "procedures." If you look in the early portions of the program you'll see a lot of GOSUBs. These route the program execution to the line specified. The program will continue along until it "sees" a RETURN, at which point it will go back to the line following the GOSUB command.

Line 31 of the structured version of the program uses a GET statement to hold the display until a character is entered. Each GET statement can be used to take as many as 10 characters from the keyboard and assign them to variables. For instance, to get the first five characters you would use a statement similar to "GET A$, B$, C$, D$, E$." The "$" following each letter variable means to treat the variable as a character value, rather than a numeric or integer value.

In line 31, I was only interested in using the first character, so I only assigned one variable. By comparing it with nothing (IF Z$="") and repeating that line until another character is received, the copyright notice stays on the screen until a character is pressed.

The next line clears the screen, giving a clean slate for the first question to appear on. The sloppy version of the program doesn't have the "clear" statements to keep everything pretty. The way the sloppy version is written, there would have to be a dozen of these statements to make the program look as nice as the structured version. The structured version takes care of this using four lines–100, 5005, 10140, and 10240.

As you can see, the first two printed questions are followed by a GOSUB 10001. This prints the answer options and grabs the answer. Because we use the variable A%, the answer must be an integer value to be valid. (We could be fancy and include a trap to evaluate the answer and request an integer value only, but that would complicate matters too much.)

GOSUB 5000 analyzes the answer to make certain that it falls between one and four, and if not clears the screen and prints the invitation to pick an appropriate answer. Execution returns to the last question section, where the next line again evaluates whether the answer was valid. If the answer is invalid, the question is repeated on the newly cleared screen, right below the request for a valid answer. If the answer was okay, a GOSUB sends the program off to evaluate the latest answer and add it to the total.

The subroutine at 10050 handles the answers for questions four and five. Since all the rest of the questions have different answers, each is followed directly by its options. The results section of the structured program is no different from the sloppy version.

TEST RESULTS

Of course, this test has not been scientifically verified or researched. I invented all the questions and answers and determined how the final evaluation went, so don't put too much weight on your test results. What the program does do, though, is illustrate how you can create quizzes for your family and friends.

It is much easier to modify a structured program than a slipshod program. Here's a challenge. Choose a group of questions that have only one correct answer and three invalid ones. Write the PRINT and INPUT statements for your questions. That's the easy part! Now, rewrite the subroutine at 10100 to lock out the incorrect answers and only give points for a correct answer.

Can you imagine how difficult it would be to go through and modify the evaluation sections for each question in the unstructured version? You'd be at it for hours. With the structured version it can be done easily.

SIMPLE AND NOT SO SIMPLE

The easiest way to redesign the structured program would be just to reassign the "T=T+?" values to 0 for the noncorrect answers and 1 for the correct answer. However, this would require six lines for each possible answer. Since each question has a possibility of four correct answers, we'd need four subroutines to cover all possible answers. After all, in a multiple choice test we don't want all the answers to be the same, right? This would mean 24 lines to cover all possibilities and one GOSUB per question to send execution to the subroutine that has the correct answer. Can you think of a way to do the same thing, but use fewer lines?

```
20100 IF A%=1 THEN T=T+1
20110 PRINT"[CLR]"
20120 RETURN
20200 IF A%=2 THEN T=T+1
20210 PRINT"[CLR]"
20220 RETURN
20300 IF A%=3 THEN T=T+1
20310 PRINT"[CLR]"
20320 RETURN
20400 IF A%=4 THEN T=T+1
20410 PRINT"[CLR]"
20420 RETURN
```

If 1 is the correct answer use GOSUB 20100, 2=20200, 3=20300, 4=20400.

Half as many lines! Not bad, huh? Actually, I showed you the difficult way first. I hope this gives you some ideas for projects of your own. If any of you would like to share some of your projects with me, I'd be delighted to take a look at them. I'm available on Viewtron (user ID 266399CCP) or through letters forwarded to me by the *Ahoy!* staff. I occasionally stop in at *Ahoy!'s Port of Call* on PlayNET as well.

Unless something more pressing comes up, I will be examining some copy products next month. If there's anything in particular you'd like to see in my column, let me know. I'm always open to suggestions. □

SEE PROGRAM LISTINGS ON PAGE 100

COMMODARES

PROGRAMMING CHALLENGES

By Dale Rupert

Each month, we'll present several challenges designed to stimulate your synapses and toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *Ahoy!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or most unusual solutions. Be sure to identify the *name* and *number* of the problems you are solving. Also show sample runs if possible, where appropriate. Be sure to tell what makes your solutions unique or interesting, if they are.

Programs on diskette are welcome, but they must be accompanied by listings. You must enclose a stamped, self-addressed envelope if you want any of your materials returned. Solutions received by the middle of the month shown on the magazine cover are most likely to be discussed, but you may send solutions and comments any time. Your original programming problems, suggestions, and ideas are equally welcome. The best ones will become *Commodares*!

PROBLEM #26-1: APPLES ANYONE?

Ted Grondski (Springfield, MA) sent a selection of problems he has proposed to his user's group. Here's a good problem in logical deduction. Ted suggests you might need access to an Apple computer to solve it, but I'll get the Commodore computers will do as well.

Eight teenagers divided 32 apples as follows: Ann got one, Beth took two, Kate three, and Dot four. Ed Smith took as many as his sister, Fred Brown twice as many as his sister, George Black three times as many as his, and Jack Robinson four times as many as his sister. What are the full names of the girls?

PROBLEM #26-2: WHAT'S YOUR SINE?

John Prager (Bay City, MI) sent a collection of mathematical problems. This one requests an angle in radians, and the computer responds with the sine of that angle. One catch, of course. Your program must not use any trigonometric or logarithmic functions.

PROBLEM #26-3: DUAL VOWELS

Write the shortest program which prints every possible combination of four letters containing any two (and only two) of the vowels a, e, i, o, or u. "Good dare" are two such combinations.

PROBLEM #26-4: SPELLED NUMBER

Is there any number equal in value to the sum of the letters of its name, assuming that A=1, B=2, and so forth? For example, "one" equals "15 + 14 + 5" or 34, but obviously "1" does not equal "34." This one shouldn't be too difficult if you refer to the solutions to *Problem #20-2: Numeral Converter,* discussed in the December 1985 column.

We have some interesting and useful solutions to the October 1985 *Commodares,* including what just might be the world's longest listable BASIC program line. We promised in October that we would list Jim Speers' (Niles, MI) solution to his *Problem #22-1: Making Change,* so here it is.

```
1 REM  COMMODARE #22-1 :
2 REM         MAKING CHANGE
3 REM  SOLUTION  BY  JIM SPEERS
4 REM
100 INPUT"ENTER AMOUNT: $";T:T%=T*100+.5
:IFT%>1000THEN100
110 INPUT"ENTER # OF COINS:";C%:X%=T%-C%
:IFX%<0THEN160
120 Y%=X%/4:D%=X%-4*Y%:Y%=Y%-2*D%:IFY%<0
THEN160
130 Q%=Y%/6:N%=Y%-6*Q%:P%=C%-Q%-D%-N%:IF
P%<0THEN160
140 Z%=Q%:Z$="QUARTER":GOSUB170:Z%=D%:Z$
="DIME":GOSUB170
150 Z%=N%:Z$="NICKEL":GOSUB170:Z%=P%:Z$=
"PENNY":GOSUB170:END
160 PRINT"NO SOLUTION POSSIBLE":END
170 PRINTZ%;Z$;:IFZ$="PENNY"ANDZ%<>1THEN
PRINT"+IES":RETURN
180 IFZ%<>1THENPRINT"S":RETURN
190 PRINT:RETURN
```

The problem was to find a combination of a specified number of quarters, dimes, nickels, and pennies which added up to a specified amount of money. We arbitrarily limited the money amount to $10.00. Jim did some algebra before writing his program. If you are interested in his solutions to the set of simultaneous equations, send me a stamped, self-addressed envelope with your request clearly stated.

The solution from B. A. Zidovec (Kitchener, ONT) lets the computer step through all possible combinations of coins, looking for the right ones. Eventually all solutions are found, but eventually can be a long time. Perhaps you can modify this program so that the current values of coins are displayed as an indication that the computer really is thinking.

```
1 REM  COMMODARE #22-1 :
2 REM          MAKING CHANGE
3 REM  SOLUTION  BY  B. ZIDOVEC
4 REM
10 INPUT"AMOUNT $";A:A=A*100:INPUT"# OF
COINS";C:PRINT:PRINT" Q D N P":PRINT
20 FORQ=0TOA/25:FORD=0TOA/10:FORN=0TOA/5
:FORP=0TOA:IFQ+D+N+P<>CTHEN40
30 IF 25*Q+10*D+5*N+P=A THEN PRINTQ;D;N;
P:F=1
40 NEXT P,N,D,Q:IF F=0 THEN PRINT"NO SOL
UTION"
```

Paul T. Dawson (Waitsfield, VT) sent a very compact and speedy solution which prints all possible coin combinations.

```
100 REM  PROBLEM #22-1 : MAKING CHANGE
110 :
120 REM  PAUL T. DAWSON
130 :
140 INPUT "AMOUNT OF MONEY (CENTS)";X
150 INPUT "NUMBER OF COINS";Y
160 FOR Q = INT((X-Y)/24) TO 0 STEP -1
170 FOR D = 0 TO 100
180 N = (X-Y-(24*Q)-(9*D))/4
190 P = Y - (Q+D+N)
200 IF Q>=0 AND N>=0 AND P>=0 AND N=INT(
N) THEN PRINT "Q"Q"D"D"N"N"P"P
210 IF N > 0 THEN NEXT D
220 NEXT Q : PRINT "FINISHED"
230 P = Y - (Q+D+N)
```

Paul's program tells you when it is finished. The amount of money must be entered in cents, although you may easily modify that if desired.

Problem #22-2: Simple Comparison brought several types of solutions. Three representative approaches have been combined into the following program.

```
1 REM  COMMODARE #22-2 :
2 REM          SIMPLE COMPARISON
3 REM  SOLUTIONS BY
4 REM    NORM GREEN (LINES 100-130)
5 REM    FRANK SMITH (LINES 200-250)
6 REM  & DENNIS FURMAN (LINES 300-350)
7 REM
10 DATA 1,2,3,4,5,6,7,8,9,10
20 DATA 10,9,8,7,6,5,4,3,2,2
30 FOR N=1 TO 10 : READ A(N) : NEXT
40 FOR N=1 TO 10 : READ B(N) : NEXT
50 CT=CT+1
60 ON CT GOSUB 100,200,300
70 IF CT<3 THEN 50
80 END
99 REM==================================
100 FOR N=1 TO 10:XA=XA+A(N):YA=YA*A(N)
110 XB=XB+B(N) : YB=YB*B(N) : NEXT
120 IF XA=XB AND YA=YB THEN PRINT : PRIN
T"MATCH" : RETURN
130 PRINT"NOT A MATCH" : RETURN
199 REM==================================
200 FOR C=1 TO 10:FOR D=C TO 10:IFA(C)<>
B(D) THEN 220
210 H=B(C):B(C)=B(D):B(D)=H:GOTO 240
220 NEXT D
230 PRINT"THE TWO SETS ARE NOT THE SAME"
:RETURN
240 NEXT C
250 PRINT"THE TWO SETS CONTAIN THE SAME
NUMBERS":RETURN
299 REM==================================
300 FOR I=1 TO 10:FORJ=10 TO I+1 STEP-1
310 IF A(J)>A(J-1)THENC=A(J-1):A(J-1)=A(
J):A(J)=C
320 IF B(J)>B(J-1)THEND=B(J-1):B(J-1)=B(
J):B(J)=D
330 NEXT J
340 IFA(I)<>B(I)THEN PRINT"NOT EQUAL":RE
TURN
350 NEXT I : PRINT"EQUAL":RETURN
```

The problem was to compare two sets of ten numbers in DATA statements to determine if the sets contained the same numbers or not. Norm Green (Perth, ONT) took an interesting approach. He calculated the sums and products of the two sets of numbers and compared those results. His program, as submitted, performed the calculations as the values were being read without having to store them in numeric arrays. Unfortunately, John's solution will not work if either list contains one particular number. Can you figure out what it is? (It is really nothing at all.)

It is also possible (but not likely) that two differing sets of numbers might have the same sum and product, isn't it? (There's another challenge for you mathematically oriented readers.) If the lists do not contain zero, and if they contain small enough numbers so that an accurate product can be determined, John's method is a quick and efficient approach.

Jim Speers also checked the sums of the numbers, but simply as a preliminary checksum. If their sums differ, Jim knows right away that the sets don't match. If their sums agree, then Jim's program performed a sort and comparison.

Frank T. Smith (Wilmington, DE), in part two of the program above, took each number in the first set and looked for a match in the second set. If a match is found, his program switches numbers in the second set to protect the matched numbers from any further comparisons. Several readers overlooked the important point of elimi-

nating any matched values from further checking. For example, the sets 1,1,2 and 1,2,2 will match unless the used values are somehow "crossed out." James Borden (Carlisle, PA) replaced the matched values with 1E38. Paul Dawson converted the numbers to strings and replaced matches with the null string "".

One other approach is exemplified by the third part of the program above from Dennis Furman (Edwards, CA). Dennis' program sorts the two sets of numbers from high to low. It compares values after each pass through the bubble sort, and it exits when the first inequality is reached or when the ten numbers are equal. You can compare the speeds of the three programs for yourself by adding timing statements such as these:

```
45 TI$="000000"
65 PRINT TI-TM "JIFFIES FOR PART" CT : T
M=TI
```

Other readers had solutions similar to the examples listed above.

Problem #22-3: Measured Decimals brought quite a few solutions. The program from Matt Shapiro (Fort Lee, NJ) listed below is representative of many of the responses:

```
1 REM  COMMODARE #22-3 :
2 REM          MEASURED DECIMALS
3 REM  SOLUTION BY MATT SHAPIRO
4 REM
10 INPUT"HOW MANY INCHES (DECIMAL)";D
20 W=INT(D):N=INT(64*(D-W)+.5):D=64:IF N
=64 THEN PRINT W+1 : END
30 N2=N/2:IF N2=INT(N2) THEN N=N2:D=D/2:
GOTO 30
40 PRINT W;"AND";STR$(N);"/";MID$(STR$(D
),2):END
```

The user enters a number of inches in decimal form, and the program responds with a whole number and a fractional number of inches rounded to the nearest sixty-fourth of an inch. Line 20 of Matt's program converts the decimal fraction to a proper fraction. Line 30 reduces the fraction to its lowest terms by repeatedly dividing the numerator by two, since two is the only prime factor of the 64 in the denominator. Notice how Matt eliminated the leading and trailing spaces normally associated with printed numerical values. The secret is to convert the numbers to strings first as he did in line 40.

Several readers sent general base conversion programs in response to *Problem #22-4: Ternary Turnabout.* The problem was simply to convert a base-ten number into its ternary (base-three) equivalent. The solution below from David Hoffner (Brooklyn, NY) is about the most compact one we received.

```
1 REM  COMMODARE #22-4 :
2 REM          TERNARY TURNABOUT
3 REM  SOLUTION BY DAVID HOFFNER
4 REM
5 B=3 : F=B+1
10 INPUT"DECIMAL";D
20 C=INT(D/B):E=INT(F*(D/B-C)):A$=RIGHT$
(STR$(E),1)+A$:IF C THEN D=C:GOTO 20
30 PRINT"BASE"B": "A$
```

As David pointed out, this program is easily modified to convert numbers from base ten to any base from two through nine. Simply change the value of B in line 5. It is also possible to convert to bases from 11 to 99 by changing the "1" in line 20 to "2". If B is 16, the conversion will give a hexadecimal value. Digits greater than 9 will be displayed as two digit numbers rather than as the letters A through F, but the results are easy to interpret.

Congratulations and thanks to the following readers who have not been mentioned earlier for their solutions to *Commodares:* George Churinsky (Denver, CO), Eric Biberhofer (Dundas, ONT), Wallace Leeker (Lemay, MO), Paul D. Hanus (Akron, OH), Joseph T. Sacco (North Bergen, NJ), Tom Hall (Madison Heights, MI), Thomsen Fung (San Diego, CA), Charles Grady (Cleveland, TN), Peter Troy (Casco, ME), Maurice Tift (Albany, GA), Richard B. Snyder (Chelmsford, MA), John Fraleigh (Wappingers, NY), and Lorne P. Korczak (Hamilton, ONT).

We all know that the longest program line that can be typed is 80 characters. Since BASIC tokenizes keywords

and provides shorthand entry of keywords, it is possible to create program lines which, when LISTed, are much longer than 80 characters. Their only disadvantage (besides unreadability) is that they can't be edited. Jim Speers suggested typing a line number followed by as many "?:" as possible. When such a line is listed, of course, you get a few dozen PRINT statements, separated by colons. In fact, such a line expands to a grand total of 241 characters, including the line number.

But you haven't seen anything yet. Donald H. Graham (Baltimore, MD) suggested ignoring the colons and simply typing 79 question marks after the line number. When you list this line, you will see a syntactically error-ridden string which totals 397 characters. Taking us even further is the suggestion from Lorne P. Korczak (Hamilton, ONT) to type a line number followed by REM and 76 shift-x characters. The shift-x's expand to the longest BASIC keyword, RESTORE. Now we're up to a grand total of 537 characters in a single program line.

Not to be outdone, Donald Graham comes back with the following statements which must be typed in direct mode (without line numbers):

```
1 REM
2 REM  LONGEST LISTABLE LINE
3 REM  BY DONALD H. GRAHAM
4 REM
5 REM THE FOLLOWING STATEMENTS MUST
6 REM BE ENTERED IN DIRECT MODE!!
7 REM
10 REM POKE46,10 : CLR <RETURN>
20 REM POKE2050,9:POKE2051,0:POKE2052,0
<RETURN>

30 REM FORI=2053TO2302:POKEI,140:NEXT:PO
KE2303,0:POKE2304,0:POKE2305,0<RETURN>
```

The result of your effort is a LISTable BASIC line containing 1752 characters. Donald agrees it is not worth much to BASIC, but perhaps you can figure out a use for such a line. Donald says that he will be glad to explain what's going on here if you write to him at 125 Cedar Hill Road, Baltimore, MD 21225.

Well, we can't quit these shenanigans without a final word from James Borden (Carlisle, PA). James says to load up your machine language monitor program and try this. Beginning at memory location 0800 (use 1C00 for the Commodore 128), enter 00 05 09 (use 1D instead of 09 for the C-128) 00 F0, and then enter 8C's up to address 0904 (1D04 for the C-128). Finish up with at least three values of 00. This gives a BASIC program line containing 1770 characters, according to James. No doubt the people at Guinness have already contacted James about this! Until next month, keep those keyboards humming. □

ALIGNING THE 1541 DISK DRIVE

Including *1541 Alignment Check* for the C-64 and 1541

By Greg Granville

Illustrations accompanying this article are reprinted from *Troubleshooting and Repair of the Commodore 1541 Disk Drive* by Michael G. Peltier, published by Howard W. Sams & Co., Inc., Indianapolis, IN, and appear courtesy of Michael G. Peltier and Howard W. Sams.

The 1541 is a very powerful disk drive for the price. Consider its unique disk formatting system which allows more data to be squeezed onto the disk by placing more sectors on the outer tracks. How about the fact that the Disk Operating System (DOS) is on a ROM chip, ready to obey our commands from the minute we turn the power on? And don't forget that our drive can support relative file structures, something that many other, more expensive drives can't do.

Despite all of the 1541's advanced features, it has at least one serious flaw (most things do): a tendency to go out of alignment far too often. The symptoms of misalignment usually include some combination of the following:

1) The red light blinks quickly a few times during a program load.

2) The drive head bumps against the track 1 stop several times and the load fails to complete. After the drive stops, the red light continues to blink, indicating an error condition.

3) The drive mechanism "clicks" repeatedly when a program load causes the drive to try to read track #1 and the track #1 stopper is improperly set.

If your drive is still under warranty when these symptoms occur, you can usually return it to your dealer for a new one. If your drive is out of warranty, you often do not have any alternative but to box up your 1541 and take it to an authorized service center for repair (get out your checkbook)!

The program that accompanies this article (see page 114) will enable you to accurately determine the alignment status of the drive yourself. You can also use it as a guide in realignment the drive, if you have the inclination to try the job on your own. Before proceeding, you may want to refer to Morton Kevelson's review of CSM Software's *1541 Disk Drive Alignment Program* (January '86 *Ahoy!*, page 84), wherein he provides an explanation of the causes and symptoms of 1541 disk drive misalignment.

THE PROGRAM

Type in the *1541 Alignment Check* program and test it out on a formatted disk. I recommend that you use an expendable disk in case you have made a typing error while entering the program. If the program is working properly, the drive motor will run as soon as the program is run. The only way to stop the drive motor and exit the program is to press 'X' on the keyboard. You should be able to move the drive head ½ step at a time by moving the joystick (port 2) to the left and right. Pressing and holding the fire button should cause the red light on the drive to flash at about one-second intervals. Hold down 'I' on the keyboard and the head should initialize back to track 18.

Now that we know that the program is working properly, we can use it to check out the alignment. You now need a disk that was formatted on a drive that is in proper alignment. I recommend using a factory formatted disk, such as the Test/Demo disk that came with your disk drive. Insert that disk into the drive and run the *1541 Alignment Check* program. Move the head (with the joystick) to track 17.5 (or any other ½ track).

After moving the head to track 17.5, press and hold the fire button (or depress the SHIFT/LOCK key) for about 10 to 15 seconds. If your drive is in close alignment you will get the message 'ALIGNMENT OK' several times during the check. Take note of the track number read from the disk. If you get track numbers both ½ step above *and* below the track the head is actually on, along with some 'ALIGNMENT OK' messages, your drive is in almost perfect alignment. A good example would be reading alternating 17's and 18's when the actual head position is 17.5. This means that your head is exactly halfway between tracks, which is right where it should be!

You can move the head around to other areas of the disk now and try reading the drive status if you like. When checking the status on full tracks, the only message you will get will be 'ALIGNMENT UNCERTAIN' unless the

alignment is way off. If a full track simply can't be read, or if the track number read from disk and the actual head position do not match, an 'ALIGNMENT BAD' message will be displayed. If the track numbers are not in sync, your alignment is off by at least one whole track.

Keep in mind that you will normally be using the half tracks to do the fine checks on the alignment. If you can get the 'ALIGNMENT OK' message to come up within 10 to 15 seconds on a half track, the alignment is usually close enough. You can perform this check on any half track from 1.5 to 34.5 in order to get the 'ALIGNMENT OK' message.

If you discover that your fears are confirmed and your drive really is out of alignment, then you have one of two choices. First, you could be content with the knowledge that you have just gained and leave the job to someone else. Or, you can be brave and attempt the job yourself. I am assuming that your drive is out of warranty; otherwise you would be more foolish than brave to tear it apart.

REALIGNING THE DRIVE

Turn the drive upside down and remove the four Phillips head screws that hold the case together. Set the drive right side up again, remove the top cover, and set it aside. Locate the two screws that hold the metal shield in place over the main circuit board. Remove these screws and lift the metal shield from the circuit board. Some older drives have a wire running through a hole in the shield, so make sure that you unplug it from the circuit board before pulling the shield off.

You now want to locate and remove six Phillips screws that hold the bottom plastic case to the main steel chassis of the drive. There are three of them down along each side of the drive chassis. You will have to unhook the wire that goes to the red L.E.D. light on the bottom plastic case. Unplug this wire from the main circuit board and leave it off until you are ready to put everything back together again.

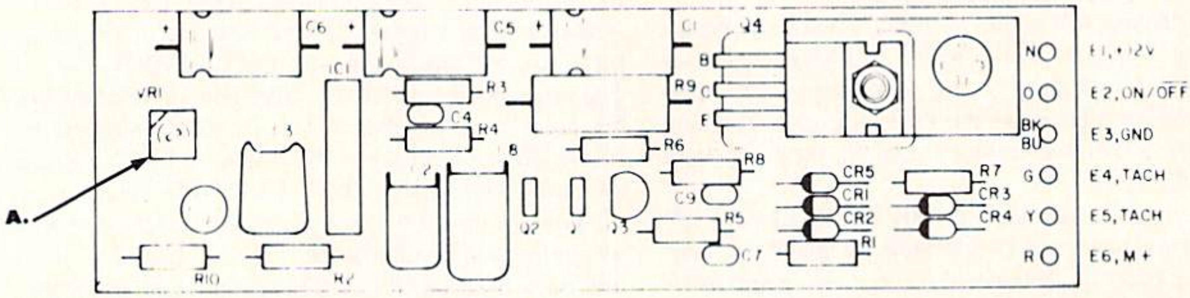
Before we remove any more wires, you might want to get a notepad so that you can write down where the wires plug into. You will feel more secure when you are putting the wires back on if you draw yourself a little diagram.

Now you can go ahead and unplug the rest of the wires from the circuit board. After the wires are removed, take out the screws that hold the circuit board to the chassis frame. Please note that there are two screws on the right rear of the drive chassis that go in from the side. Now that you finally have everything loose, you can carefully remove the main circuit board.

Exposed to your eyes now are the mechanical innards of the drive. As you face the unit from the front you will see the black head assembly in the center area. Note that the head assembly rides on two shiny metal rails. You can gently slide the head back and forth on these rails if you like.

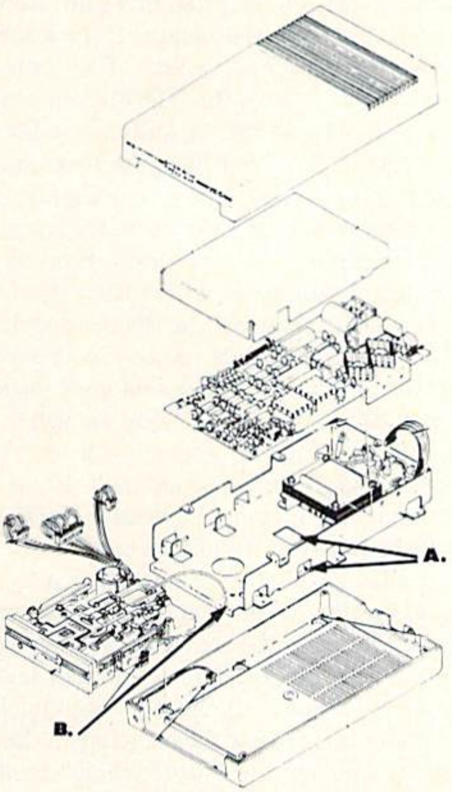
If you have some Vaseline or a light silicone grease, putting a very small quantity of it on each of the rails will make the drive run quieter while it is stepping the head. Be sure that you don't get the grease on anything else except the guide rails. It is also a good idea to slide the head back and forth after greasing the rails to work it in a bit.

Slightly to the right of the head assembly you will see two pulleys and a thin steel band connected between them. Note also that the head assembly is connected to this band. When the pulleys turn, the band moves, and the head assembly also moves. The pulley in the front is only there to keep the band stretched tight. Note that it has a spring connected to it, so that constant tension is being applied. The pulley that you see in the rear is the stepper motor pulley itself! You want to be certain that this pulley is tightly pressed onto its shaft. Take your Phillips screwdriver (or any blunt-ended tool) and place it on the top surface of the pulley (not the shaft). Now hit the top of the screwdriver with your hand to firmly seat the pulley into place. If you saw the pulley move downward even the smallest bit, then it was loose. Just for good measure, carefully hit it again on the opposite side to make sure that it is squarely in place.



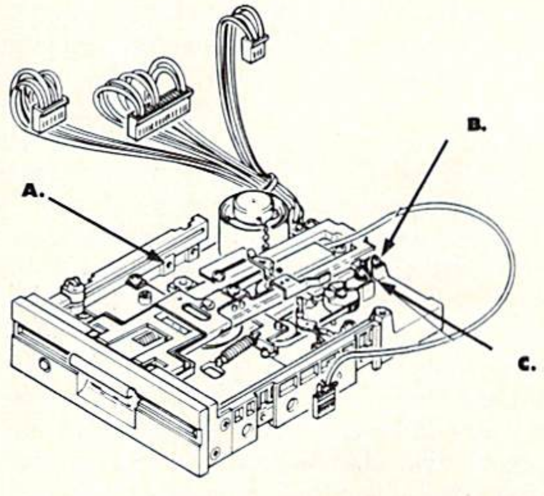
DRIVE SERVO CIRCUIT

A. Drive speed adjustment
Drive servo circuit is located under drive mechanism in 1541's equipped with ALPS drive units. It is located above mechanism on 1541's using the Nortronics drive mechanism. The Nortronics drives have a twist latch lever on the disk drive door.



1541 DISASSEMBLY/REASSEMBLY

A. access holes to stepper motor locking screws
B. access hole for drive speed adjustment



TRACK #1 STOP ADJUSTMENT

A. Drive servo circuit located here on Nortronics drives.
B. Track #1 stop
C. Lock down screw

Once you have the stepper pulley tightly pressed onto the shaft, you might want to apply some five-minute epoxy to the top of the pulley and shaft to prevent any possible slippage from occurring in the future. Put the epoxy in the small gap between the two protrusions on the top of the pulley that is opposite the track one stopper. Make sure that some of this epoxy runs up onto the top surface of the shaft itself, so that the bond will interconnect the pulley and the shaft. Be certain that none of it gets in the wide gap area where the track one stopper protrudes. Keep an eye on the epoxy as it cures to make sure that none of it seeps into this area. If it does, remove it before the epoxy dries completely.

With all our preventive maintenance out of the way, we can now go ahead and adjust the alignment. Mount the circuit board back onto the drive chassis and reconnect all the wires you removed. (Don't worry about the wire that goes to the red L.E.D. on the bottom case; you can leave it off for now.) Set the drive on its right side so that you can get at the bottom of it. Locate the two

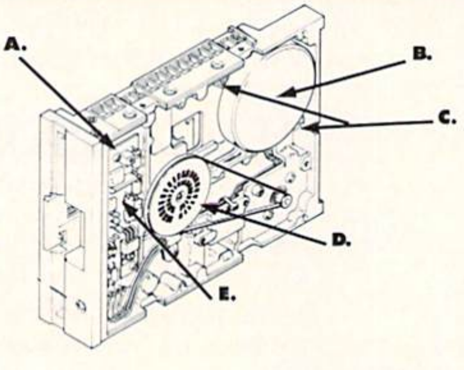
CAUTION!

Readers should be aware of the possible hazards that exist when working inside sensitive electronic equipment. If you do decide to follow the procedures described here you are doing so at your own risk.

Power line voltage (120 volts) is present in the 1541 disk drive. This can present a hazard to both personnel and equipment. Contact of the drive circuit board with the metallic chassis at any point other than the designated mounting points may damage the drive electronics. The same is true for metallic tools that contact the exposed circuit board or its connector.

openings about halfway back on the metal chassis that each have a Phillips head screw visible through them. These screws hold the stepper motor in position.

Power up your system and load the *1541 Alignment Program* so that we will have it ready. Now go ahead and loosen the two stepper motor screws. The screw holes are slotted so as to allow about ⅜ inch of movement of the entire stepper motor. Once it is free for adjustment, run the program with the Test/Demo disk inserted. Move the head to a full track (try 17) and press the fire button (or shift lock) to read the actual head position. You can now turn the stepper motor through the holes with your fingers. You want to get the position read from disk to say track 17, with no read errors indicated. Once you have that, move the head to track 17.5 and press down the SHIFT/LOCK button so that both your hands are free. This is the fine adjustment, so you may have to move the motor only a very small amount at a time. With the actual head position of 17.5 try to get the position read from the disk to be on the edge of jumping back and forth from 17 and 18. When you are close to being exactly on the half track, you will probably get a lot of "Read Error Sensed" and "Alignment OK" messages. If you can't get the track read from the disk to jump back and forth from 17 to 18 while reading the drive status, then try to get as many read errors as possible along with an occasional track read of the lower number (track 17). Many drives will show almost nothing but "Alignment OK" messages when they are exactly on the half track, since there is nothing on the half track for them to pick up. Different areas of the disk may not give exactly the same results on the half tracks, but as long as you are getting some "Alignment OK" and "Read Error Sensed" messages, you are OK.



DRIVE UNIT, PARTS LAYOUT (BOTTOM)

A. Speed adjustment (VRI)
B. Stepper motor
C. Locking screws
D. Strobe markings for speed adjustment
E. Drive motor servo circuit (located on top of Nortronics drive mechanism)

Once you think that you have it as close as you can get it, tighten down the screws on the stepper motor and run the check again. Do this to be sure that you did not bump anything while you were tightening the screws. If everything still looks good, turn the drive off and let's verify everything from a cold start just to be sure. Turn the drive back on and run the program again. Move the head to track 17 and press the button. If the track read from the disk is 17 everything is fine; if not, you must loosen the screws and make the adjustments again.

We're almost done now, but we still have a few little things that we should check out. We must make sure that the track one stopper is set so that it is just barely touching when the head is on track one. Move the actual head position to track one with the program. Stop the program by pressing 'X' and remove the main circuit board again. Examine the gap between the stopper and the protrusion on the stepper pulley. If you can see a gap here, then loosen the one Phillips screw that goes through it and move the stopper closer. Move it just enough to eliminate the gap at this point. If you move it too far and push on the pulley itself, it will be too tight. If you heard a thud when you were moving the head to track 1, you may have to open up a very small gap between the stopper and the pulley. (Ed. note: We found that a gap of .006 inch works well with the 1541 drive.) After you adjust the stopper, reconnect the circuit board. Run the program, and move the head position back to track one. Listen for the thud noise just as the indicator says track one. Turn off the power and remove the board again. Recheck the gap to see if you have the "just touching" condition. If you have to, go ahead and repeat the above procedure again until you get it just right.

If you are *very* careful that the circuit board does not touch the chassis, you can avoid having to remove it each time you want to check your adjustment. Just leave the wires connected and move the board off to the side so that you can get to the stopper adjustment. Placing a piece of thin cardboard between the board and the chassis will help in preventing an accident.

You may still hear a faint thud when the adjustment is correct. Just make sure that the drive can read track one. Press the fire button to see if you can read it with no errors.

Take a minute to examine the strobe pattern that is visible through the large hole in the bottom of the drive. Under a fluorescent light, the outer pattern should appear to stand still or drift very slowly while the drive is running. If it does not, the speed of the drive motor is improper. It can be adjusted with a small screwdriver through a small hole that is toward the front left side of the drive bottom. (This adjustment is made on top of the drive in 1541's equipped with Nortronics mechanisms.) Problems in the motor speed servo circuit seem to be quite rare.

The job is now completed! Put the rest of the screws back in and put the shield and case back on. Remember to hook up the wire to the front red L.E.D. again. Your reward is a drive that runs both smoother and quieter and, best of all, you have just saved yourself a pile of money. Congratulations on a job well done! ☐

SEE PROGRAM LISTING ON PAGE 114

TIPS AHOY!

Compiled by Michael R. Davila

***Tips Ahoy!* is not founded on the democratic principle. That's why you'll see the same names popping up month after month. We're not interested in getting everyone involved—just in publishing the most original and useful programming tips available from the Commodore community. If you have a tip that fits this description and want to earn up to $50 (or more), send it to *Tips Ahoy!*, Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001.**

PARALYZED CURSOR

It's often undesirable to have the person using your program able to move the cursor around the screen when an INPUT statement is used. There are a number of ways to deter this; here are two examples.

The simplest way is to POKE 198,1:POKE 631,34 before each INPUT. The first POKE tells the program that a key has been pressed, and the second puts a quotation mark in the keyboard buffer. This forces the computer into quote mode, and the cursor keys will appear as reverse graphics characters. You're not really disabling the keys, just preventing them from moving the cursor around the screen.

Another method is to use a short machine language program that completely disables the cursor, both inside and outside the program. Try the one below.

```
10 FOR A=710 TO 758:READ B:POKE A,B:NEXT
20 DATA 169,229,141,143,2,169
30 DATA 2,141,144,2,173,246
40 DATA 2,73,1,141,246,2
50 DATA 208,10,169,72,141,143
60 DATA 2,169,235,141,144,2
70 DATA 96,165,203,201,7,240
80 DATA 4,201,2,208,4,169
90 DATA 64,133,203,76,72,235,0
```

SYS 710 turns off the cursor keys if they're on, and it also turns them back on if they're disabled. It's quite effective, and you don't have to worry about properly formatting POKE statements to do the job. With this method you avoid the reverse characters, which is both convenient and effective.

—Kevin Brown
Anderson, IN

DEFUNCTIONED FUNCTION KEYS

Commodore 64 BASIC programs may be loaded and saved in 128 mode, but may not include BASIC 7.0 commands. The following routine restores the 128's function keys to their state when in 64 mode. Also, this is done without implementing the KEY command. The routine must be executed in 128 mode.

—Ed Horgan
Coatesville, PA

```
1000 A$="133137134138135139136140"
1010 J=1:FORI=4106TO4113
1020 POKEI,VAL(MID$(A$,J,3)):J=J+3
1030 POKEI-10,1:NEXT:A$=""
```

INSTANT DISK ERROR READ

The next time your disk drive light starts beating at 100+ BPM (Blinks Per Minute), simply press CTRL and D. This will instantly display what your drive's trying to communicate to you.

```
10 REM DISK STATUS
20 REM BY SHAWN K SMITH
30 R$=CHR$(18):PRINT,R$"DISK STATUS"
100 DATA 234,120,169,075,162,003,141,020
110 DATA 003,142,021,003,088,096,173,141
120 DATA 002,201,004,208,006,032,091,003
130 DATA 032,062,003,076,049,234,169,088
140 DATA 162,003,120,141,020,003,142,021
150 DATA 003,088,165,203,201,018,240,001
160 DATA 096,169,008,032,180,255,169,111
170 DATA 133,185,032,150,255,169,000,133
180 DATA 144,032,165,255,032,210,255,165
190 DATA 144,240,246,076,171,255
195 :
200 FOR M =829 TO M + 77:READ Y:POKE M,Y
210 T=T+Y:NEXT:T=T-8849
220 IFTTHENPRINTR$"**DATA ERROR**":STOP
230 POKE857,PEEK(788):POKE858,PEEK(789)
240 SYS830
250 PRINT:PRINTR$"CNTRL-D DISPLAYS STATU
S..."
260 PRINT:PRINTR$"RUN/STOP-RESTORE DEACT
IVATES"
270 PRINT:PRINTR$"SYS(830) WILL REACTIVA
TE"
```

The utility wedges into the IRQ interrupt and can be disabled with RUN/STOP-RESTORE. A SYS to the starting address (830) will reinstate the wedge.

Since this utility wedges into the "Hardware Interrupt," it will perform in direct or program mode.

—Shawn K. Smith
Bronx, NY

DIRECTORY HIGHLIGHTER

Here is a handy disk utility that allows you to highlight any program name in the directory. This is useful in making certain programs stand out, such as program boots. The highlighting is achieved by renaming the file with four special characters preceding it. The four characters are shifted space, two delete characters, and a reverse character. This leaves enough space for only twelve

characters for the rest of the filename. The routine below will first ask you whether you want to highlight or de-highlight and then ask for the filename. To end the program reply to the highlight/de-highlight prompt with any key other than 'H' or 'D'.

```
5 OPEN 15,8,15:GOSUB 1000
10 SS$=CHR$(160):DL$=CHR$(20):RV$=CHR$(1
8)
20 HD$=SS$ + DL$ + DL$ + RV$
30 PRINT"[CLEAR][RVSON]H[RVSOFF]IGHLIGHT
 OR [RVSON]D[RVSOFF]E-HIGHLIGHT PROGRAM"
40 GET KY$:IF KY$=""THEN40
50 IF KY$="H" THEN 100
60 IF KY$="D" THEN 200
70 END:REM END PROGRAM IF NOT 'H' OR 'D'
100 INPUT "PROGRAM NAME TO HIGHLIGHT";HP
$
110 PRINT#15,"R0:" + HD$ + HP$ + "=" + H
P$:GOSUB 1000
120 GOTO 30
200 INPUT "PROGRAM NAME TO DE-HIGHLIGHT"
;DP$
210 PRINT#15,"R0:" + DP$ + "=" + HD$ + D
P$:GOSUB 1000
220 GOTO 30
1000 INPUT#15,E,E$,T,S
1010 IF E=0 THEN RETURN
1020 PRINT E,E$
1030 END
```

Note: To load a highlighted filename in either program or immediate mode you can use LOAD"(shifted space)??? filename",8,1 *—Dino Bavaro*
Don Mills, ONT

EASY BACKUPS

Here's a reliable, easy way to make sure you always have a backup to any C-128 program you're working on; it even automatically keeps *three* copies on the disk for you! (Just in case you make an "improvement" that doesn't work.)

I use the program names "CURRENT", "OLD", and "OLDEST"; you can, of course, substitute names of your own choice.

To SAVE the current version of the program (and erase the very oldest), just type GOTO 3. *—Bruce Jaeger*
St. Paul, MN

Commodore 128 Version

```
1 GOTO 10
3 SCRATCH "OLDEST":RENAME "OLD" TO "OLDE
ST" : RENAME "CURRENT" TO "OLD" : DSAVE"
CURRENT" : END
10 REM YOUR PROGRAM BEGINS HERE
```

Commodore 64 Version

```
1 GOTO 10
3 CLOSE15:OPEN15,8,15,"S0:OLDEST":PRINT#
```

```
15,"R0:OLDEST=0:OLD"
4 PRINT5,"R0:OLD=0:CURRENT":SAVE"CURRENT
",8:CLOSE15:END
10 REM YOUR PROGRAM BEGINS HERE
```

FORGOT THE SYS?

Remembering the SYS numbers for all your machine language programs can be quite tedious and annoying. Here is an easy way to find the starting address of a machine language program if you are using a disk drive.

```
10 OPEN 2,8,2,"FILENAME"
20 GET#2,A$,B$:Z$=CHR$(0):CLOSE2
30 PRINT ASC(A$+Z$)+256*ASC(B$+Z$)
```

If you wish to find the ending address, type LOAD "FILENAME",8,1 and then after the program has finished loading type PRINT PEEK(45)+256*PEEK(46) and hit RETURN.

—Wesley Vriend
Houston, BC

COLOR PROTECTOR

Many programmers have individual preferences for certain background, border, and text colors and use them when programming. Unfortunately, whenever RUN/STOP-RESTORE is pressed, the computer resets these colors to the blue default colors. Also, RUN/STOP-RESTORE will disable some ML programs such as the DOS wedge. In order to reenable the program, the appropriate SYS must be entered. Resetting screen colors and reenabling programs becomes very tiresome!

The short routine below (for the C-64) will solve these problems. From now on, RUN it before each programming session. Whenever you hit RUN/STOP-RESTORE (or simply RESTORE), the screen colors will be set to the custom that you selected by placing them in the routine. These POKEs, as you can examine in lines 30 to 60, can be POKEd with a different value at any time; after POKEing the locations, type SYS 32785 so that the routine will recognize the new values.

Each time you LOAD a new ML program type SYS 32785. Otherwise, the program will be disabled by pressing RESTORE. The routine will also allow you to recover from many more keyboard "lockups" by pressing RESTORE. A note to tape users: you must type SYS 8 to deactivate any ML programs that are enabled for a tape save to work.

—Kevin Brown
Anderson, IN

```
10 POKE 56,128: CLR: FOR K=32768 TO 3287
6: READ M: C=C+M: POKE K,M: NEXT
20 IF C <> 13765 THEN PRINT"ERROR IN DAT
A STATEMENTS":END
30 POKE 1020,03: REM- BORDER COLOR --CHA
NGE
40 POKE 1021,00: REM- SCREEN COLOR --VAL
UES
50 POKE 1022,01: REM- CURSOR COLOR --IF
```

```
YOU
60 POKE 1023,01: REM- CHARACTER COLOR --
 WISH
70 SYS 32785:PRINT"[CLR][9 DOWN]HIT REST
ORE TO ACTIVATE"
80 PRINT"[DOWN]TYPE SYS 32785 TO READ VA
LUES AFTER CHANGING THEM"
90 DATA 11,128,11,128,195,194,205,56,48,
161,128,32,17,128,76,29,128,160
100 DATA 204,185,52,2,153,128,128,200,20
8,247,96,169,0,141,252,3,169,11
110 DATA 141,253,3,169,1,141,254,3,141,2
55,3,169,58,141,0,128,141,2,128
120 DATA 108,20,3,32,129,255,32,138,255,
160,204,185,128,128,153,52,2,200
130 DATA 208,247,173,255,3,141,33,208,17
3,252,3,141,32,208,173,254,3,141
140 DATA 134,2,169,147,32,210,255,173,25
3,3,141,33,208,162,255,76,139,227
```

FASTER *FAST LOAD* DIRECTORY MENU

Have an Epyx *Fast Load* cartridge? Put this little program at the beginning of any disk. When you press the COMMODORE and RUN/STOP keys the routine will list the disk's directory and place your cursor by the first file:

```
100 PRINT"[CLEAR]$"
110 FORL=0TO5:READA:POKE631+L,A:NEXT
120 POKE198,L:END
130 DATA19,13,19,17,17,17
```

This works by using the dynamic keyboard method of fooling the computer into thinking *you're* typing all those keys. The DATA statement on line 130 holds the ASCII values of HOME, RETURN, HOME, DOWN, DOWN, DOWN in that order.

This will also function with the DOS Wedge in place instead of *Fast Load*. *—David Paulsen, Renton, WA*

TESTING FOR 64 OR 128 MODE

When programming the 128 for compatibility with the 64, the variables DS and DS$ cannot be used. These are reserved on the 128 to indicate disk status. A line like 100 DS$="DO SOMETHING" will produce a SYNTAX ERROR when executed in 128 mode.

We can use DS$ to our advantage, however, to test for operating mode. Line 10 sets M to 64 unless DS$ is other than null. If it is, M is set to 128. M can then be used as a flag to call subroutines appropriate to the 64 or 128. *—Ed Horgan, Coatesville, PA*

```
10 M=64:IFDS$<>""THENM=128
20 PRINT"SYSTEM IS IN"M" MODE"
```

ERRATA

AHOY! DOCK

Readers reported a STRING TOO LONG ERROR. To repair, change line 0 to read

```
0 PRINT "[CLEAR]": P$=CHR$(13)
```

Also, NEXT WITHOUT FOR and OUT OF MEMORY errors will emerge if you try to delete an item from the file. To correct this retype lines 7190 and 7200 to

```
7190 PRINTTAB(4);"ONE MOMENT DELETING RE
CORD #";I:I=C: NEXT:FOR I= 1 TO C
7200 IF SU$(I)="DELETE" THEN C=C-1: GOTO
 7220
```

SOLITAIRE

Due to the background color being the same as the text color, the command "TURN DECK OVER" was not visible during play. To remedy this you may change the POKE value in line 25 from POKE 53281,1 to POKE 53281,12 or any other color that you prefer.



"I told him not to mess with the copy protection on THAT disk!"

ARENA
For the C-64
By Bob Tedesco

Arena is an arcade style game for the Commodore 64 written entirely in machine language. A fast-paced shoot-em-up, it can be played at three different speed levels. *Arena* can be entered after loading by typing SYS 49152. It is played with a joystick plugged into port 1. The opening screen will ask for the speed level desired: slow, medium, or fast. Once the appropriate key is pressed (S, M, or F), the action will commence at once.

The game takes place in an "arena" represented by a square section of the screen. The arena is composed of interlocking shapes making up a grid. The player controls his 'spinner,' a whirling propellerlike object, pitted against computer-controlled opponents which move about as brightly colored geometric shapes and whose speed more than makes up for the somewhat predictable pattern of movement.

Moving the joystick up, down, left, or right will move the spinner accordingly. Pressing the joystick button while moving the joystick will enable the spinner to fire and destroy each opponent. When all opponents are destroyed, play will proceed to the next level. Points are awarded on a level-by-level basis: one point is awarded for each opponent destroyed on Level One, two points on Level Two, three on Level Three, etc.

After the game ends a new game can be started by pressing the joystick button. Typing f1 instead will bring back the opening screen so that players may change game speed (and clear the high score in the process). Pressing 'Q' will exit *Arena* and reset the computer (if exited any other way, BASIC will not function properly).

SEE PROGRAM LISTING ON PAGE 111

FLOTSAM

While we didn't ask frequent *Ahoy!* contributor Shay Addams to write the letter printed herewith, we did think his views on adventure gaming and Arnie Katz's spirited rebuttal of same would make worthwhile reading for the game fans in our audience.

You don't have to be a computer press luminary to air your views in *Flotsam*. Send your comments on any aspect of Commodore computing to *Flotsam*, c/o *Ahoy!* Magazine, Ion International Inc., 45 West 34th Street–Suite 407, New York, NY 10001.

In your November issue's "Call to Adventure," the writer seriously misinformed your readers.

First, his "history" of adventure games says they were inspired by "Dungeons and Dragons"–a popular but false legend that is often repeated by lazy writers who don't bother to research their subject. Last year I interviewed William Crowther and Don Woods, who created the first such game, *Adventure,* and both said they never played D&D until *after* their computer game was finished. So how could it have influenced them at all? (And why, in a history of adventure, were their names not even mentioned?) Also, he says Scott Adams converted *Adventure* to micro format –when Adams actually wrote an original game, *Adventureland,* that he says was inspired by the former.

An even more serious mistake was made when the writer attempted to categorize adventures from his own point of view, calling games with pictures "illustrated adventures" and *Ultima*-style games "graphic adventures." This is absurd: games with pictures are graphic adventures; those focusing on combat, magic, and character development (like *Ultima*) are fantasy role-playing games. Of course, role-playing games are considered a type of adventure, with a big difference–a difference that lies not in the visual effects, but in the nature of the playing experience.

The definition of fantasy role-playing games is best summed up in the foreword of all Penguin Software RPGs: "...a game in which the player assumes the identity of a character within the fantasy world of the game itself. Such a character is usually formed by assigning random values to special characteristics such as Strength, Wisdom, Intelligence, Luck, or Charisma. These characteristics in turn determine the capability of the character in combat, negotiation and encounters with other beings." It goes on to explain that a game's goal can usually "only be reached by nurturing a character until it is of sufficient power to overcome the most assuredly dreadful guardians blocking the path."

In a pure adventure, the player's experience consists of solving logical puzzles, not developing a character whose life hinges on the number of hit points he has in store. Certainly, you may play the role of a character in some adventures (Mac Steele in *The Serpent's Star*), but in most *you* enter an alternative setting *(Zork, Treasure Island)*–therefore, the role-playing aspect of adventure is an attribute of certain games, not of the genre itself.

These are not my opinions on the matter, but facts easily confirmed by talking with Mike Berlyn (*Suspended*), Dave Albert (*Xyphus*), or Penguin President Mark Pelczarski or anyone else in the business.

This, however, is an opinion: after reading those sketchy capsules, I can't believe the writer actually played all the games he described. How else could he have wasted space on such dogs like Screenplay's *Dunzhin* series and the primitive *Ali Baba* (a classic years ago but now obsolete and a waste of money and time)?

Apparently, this guy's research consists of reading the press release

or back of the box, not playing the game. Perhaps Mr. Katz will do a better job when mapping the maze in more familiar territory—*Ms. Pac-Man.*

—*Shay Addams*
Wayne, PA

Arnie Katz replies:

Shay Addams' letter contains a lot more heat than light. Stripped of its nose-in-the-air snideness and implications that only *Shay Addams knows about adventure games, it offers few hard facts. Worse, it makes the implicit assumption that if two people hold differing opinions, one of them is not only wrong, but Evil. The idea that there is a spectrum of opinion on every topic has apparently not occurred to him. One shudders at the thought of dinner conversation* chez *Shay.*

Let's begin with "Dungeons & Dragons." The claim that Crowther and Woods never played "D&D" before designing Adventure *doesn't mean that "Dungeons & Dragons" didn't inspire the entire field of adventure gaming, electronic and non-electronic.*

"D&D" is the most important gaming phenomenon since "Monopoly." It is inconceivable that the designers of Adventure, *with their great interest in such things, had not heard descriptions of the theory and practice of "Dungeons & Dragons." That's like the producer of the latest science fiction film about heroic rebels battling the evil empire saying that he wasn't influenced by "Star Wars." "D&D" is part of popular culture. You don't have to read a Conan Doyle story to know about Sherlock Holmes; you don't have to play "D&D" to know the rudiments of adventuring. And considering the well-known reputation of TSR (creators of "Dungeons & Dragons") for suing to protect "D&D" from even indirect infringement, what are the odds against* any *adventure author acknowledging a debt to this landmark design where TSR's legal eagles might read it?*

I'm sure Scott Adams had just this point in mind if he said that Adventureland *was not essentially Crowther and Woods'* Adventure *revised and restyled for micros. No one who has played both could miss the similarities.*

There are as many ways to classify adventure games as there are adventure gamers. Addams' approach is no worse than most, but only a self-important blowhard would imagine that it is the only possible means. Personally, I think any distinction between adventures and what Addams terms "role-playing" games is illusory. As experienced gamers know, Wizardry *and* Ultima *have no more role-playing than* Amazon *or* Mindwheel. *The main difference is that the former require the gamer to work up a character little by little, while the latter give the gamer a full-blown persona. In other words, games with "experience levels" focus on the process by which someone becomes Conan, while ones without this element concern themselves with what happens once that level of power is achieved.*

While drawing a line between illustrated and graphics adventures won't win me the Nobel Prize, I think it does address a fundamental difference. The picture in an illustrated adventure is an external object. The gamer can look, but not directly affect it. The graphics adventure, which often employs icons and symbols as well as illustration, puts the protagonist right into the scene. That's only one man's opinion, but unlike Shay Addams, I know that.

It is distasteful to have to respond to Addams' infantile challenge to my credentials. In my egotistical way, I had hoped everyone already knew me. I've been writing about entertainment software for the last eight years and was co-founder (will Bill Kunkel) and editor of Electronic Games. *The Electronic Industry Association, the computer industry trade group, has selected me to chair to Software Showcase awards committee every year since its inception. I am the co-author of* Borrowed Time, *Activision's new detective adventure and have done major development work on* Star Trek *(Simon & Schuster) and* The Fourth Protocol *(Bantam) to name just two adventure game titles. Further references on request. And I don't wash windows....*

LITTLE WINDOWS

Continued from page 23

or from the number in location 53284. (This is background color 3, and the *A* in screen memory has its two high bits set to 11, for a value of 192 added to the screen code for *A*.)

When you enter shifted strings or convert existing strings to their shifted equivalents, and then either turn on Reverse mode (CTRL-9) or turn it off (CTRL-0), BASIC will automatically PRINT those strings into screen memory with the high bits set to the code for one of the four background colors.

To turn on Extended Background Color Mode, simply:

```
POKE 53265,PEEK(53265)OR 64
```

The program *Extended Background* gives a demonstration of how to convert strings from one mode to another, and then shows how PRINTing a string with each attribute looks on the screen.

EXTENDED BACKGROUND COLORS

To give the effect of windows with different background colors, then, you merely have to make sure that every string PRINTed within a window's area has the correct combination of SHIFT/UNSHIFT and REVERSE ON/REVERSE OFF attributes.

The program *Windows* divides the screen into four areas, and lets you type whatever you like within each window. The cursor, RETURN, HOME, and CLR keys work normally–but only within the window. To change from one window to another, press f1, f3, f5, or f7. To end the program, press RUN/STOP.

You'll notice that PRINTing shifted or reversed SPACE characters defines each window by filling it with its background color.

The program is, of course, entirely useless. But it does demonstrate how you go about defining screen areas and PRINTing within them.

Each character you type is converted into a number from 0 to 63. These numbers are used as indexes into the string array CH$(3,63). This array contains the correct ASCII value for either shifted or unshifted characters.

Then the variable FC$(3) is PRINTed before each character is PRINTed into its window. FC$*(n)* consists of a foreground color character and either a REVERSE-ON or a REVERSE-OFF character, depending on which background color is needed.

Therefore the character's ASCII value in CH$(n,n) decides whether it is shifted or not, and the window's FC$(n) string decides whether it is reversed or not.

It is important that, in changing background colors from window to window, the foreground colors be

changed as well. Many foreground-background color combinations have such poor contrast that they are completely illegible.

Of course, you don't have to use Extended Background Color Mode with windowlike screen formats. As you saw in December's *Gypsy Pilot* game, the extended background colors were used more for highlighting than to define particular screen areas. As long as your game uses only the alphanumeric characters, you might as well be in Extended Background Color Mode and use the different colors to create a far more interesting screen.

CRITIQUING GYPSY PILOT

Gypsy Pilot (December) was a long one–about 150 disk sectors when it is typed in and saved. And if you've played it, you know that even though it is certainly a *complete* game, it is certainly not a *finished* game.

One of the biggest problems with it is that every now and then, as you're coming back into space after a visit from a planet, the game seems to stop and hang for a *looooong* time. Unfortunately, this is unavoidable as long as the game depends on C-64 BASIC string manipulation. Because of the clumsy way BASIC handles string formation in memory, lots of garbage is created–sections of string memory that once contained useful strings but now are used for nothing. Every string operation causes more garbage to be created. Finally, string memory is filled up. The only solution is to do "garbage collection" by issuing a command like this:

```
X=FRE(9)
```

The *X* and the *9* in that statement are dummies. They don't do anything, but they have to be there. The FRE( ) function gets rid of the garbage and tightens down string memory, then returns the amount of memory between the highest array variable address and the lowest string memory address–that's how much memory is left for your program to use. In the middle of the game, of course, we don't *care* how much memory is left. But we still have to use the FRE( ) function to get rid of the garbage. And it takes practically forever to do the job.

The solution? It's quite simple–don't use BASIC's string handling. Ideally, the game's text would be handled in machine language, by loading ranges of text into memory and then copying them into screen memory as needed. But if I had done that, none of the program would have been in BASIC, and so it would all have been incomprehensible to those of you who aren't yet into machine language. So the garbage collection is the price you pay for having the game come out in a column on BASIC game programming!

There are other things that should be done to finish the game properly. For one thing, it would be better if you couldn't leave a planet and then immediately return the ship to the same onscreen sprite and have it be a different "world" when you land! That should be fairly simple to accomplish, by not letting the player visit a planet again until it wrapped at the edge of the screen.

The on-planet sequences should have shown you the price you were being charged for each valuable item, instead of choosing cost and value randomly *after* you've already bought it.

Because the game contains only 22 different inhabited worlds and 10 uninhabited ones, it simply ends after you've visited about 30. There should have been an ending screen explaining that your term as captain has ended normally, so you don't think you made a mistake. That was simply an oversight on my part–an oversight combined with the fact that *Ahoy!* was going to press and they kind of wanted the finished program to go with my column...

More telling, however, is the fact that there is no *sound* with the game. It should have had sound. When the ship is moving, it should make one kind of sound; when fuel and supplies get low, there should be two different warning sounds. Filling up with fuel and supplies should cause another sound. Each kind of message on a planet should be introduced with a unique sound or tune. And there should have been introductory and closing music.

It could have had all that, too. But you would have had to type in another 50 sectors worth of program. There are limits to how much we can expect you to do. (You didn't know that, did you?)

Most important of all is the fact that the on-planet events are selected with a second-rate artificial-intelligence routine. The algorithms deciding what you find, and what happens to you, and how the aliens or human respond to you, are primitive at best. Some of the flaws can be solved by fine-tuning. Others require new algorithms.

All these flaws in the game are the sort of thing that would turn up in months of testing before a commercial game was released. I don't have the luxury, with a monthly column, of doing that kind of play-testing and fine-tuning–as it was, I stretched the patience of the folks at *Ahoy!* about as far as they can go before they send somebody to North Carolina to break my knees.

What's disgusting is the alarming number of commercial games that are released without having solved problems like these. You've seen some of them–you may even have bought some. And you sit there after playing a few times, wondering how in the world somebody got paid for turning out something as unpolished and unprofessional as *that.* It's one thing to find flaws in an example game that you type in from a magazine column that is trying to teach game programming techniques. It's something else to plunk down fifty bucks for a supposedly professional game that doesn't work any better.

So when you get your game program to a point where it's playable, like my *Gypsy Pilot* game, please don't think that your work is finished. You still have weeks and months of fine-tuning ahead of you before it's ready to go.

The difference is that once you have a working version, it stops being a program and starts being a game. Instead of lines and lines of code, you experience it as events on a screen. Believe me, it makes the work a *lot* easier. ☐ *SEE PROGRAM LISTINGS ON PAGE 102*

PROGRAM LISTINGS

Attention new *Ahoy!* readers! You must read the following information very carefully prior to typing in programs listed in *Ahoy!* Certain Commodore characters, commands, and strings of characters and commands will appear in a special format. Follow the instructions and listing guide on this page.

On the following pages you'll find several programs that you can enter on your Commodore computer. But before doing so, read this entire page carefully.

To insure clear reproductions, *Ahoy!*'s program listings are generated on a daisy wheel printer, incapable of printing the commands and graphic characters used in Commodore programs. These are therefore represented by various codes enclosed in brackets [ ]. For example: the SHIFT CLR/HOME command is represented onscreen by a heart. The code we use in our listings is [CLEAR]. The chart below lists all such codes which you'll encounter in our listings, except for one other special case.

The other special case is the COMMODORE and SHIFT characters. On the front of most keys are two symbols. The symbol on the left is obtained by pressing that key while holding down the COMMODORE key; the symbol on the right, by pressing that key while holding down the SHIFT key. COMMODORE and SHIFT characters are represented in our listings by a lower-case "s" or "c" followed by the symbol of the key you must hit. COMMODORE J, for example, is represented by [c J], and SHIFT J by [s J].

Additionally, any character that occurs more than two times in a row will be displayed by a coded listing. For example, [3 "[LEFT]"] would be 3 CuRSoR left commands in a row, [5 "[s EP]"] would be 5 SHIFTed English Pounds, and so on. Multiple blank spaces will be noted in similar fashion: e.g., 22 spaces as [22 " "].

Sometimes you'll find a program line that's too long for the computer to accept (C-64 lines are a maximum of 80 characters, or 2 screen lines long; VIC 20 lines, a maximum of 88 characters, or 4 screen lines). To enter these lines, refer to the *BASIC Command Abbreviations Appendix* in your User Manual.

On the next page you'll find our *Bug Repellent* programs for the VIC 20 and C-64. The version appropriate for your machine will help you proofread our programs after you type them. (Please note: the *Bug Repellent* line codes that follow each program line, in the whited-out area, should *not* be typed in. See the instructions preceding each program.)

Also on the following page you will find *Flankspeed*, our ML entry program, and instructions on its use. □

Call *Ahoy!* at 212-239-0855 with any problems.

When You See	It Means	You Type		You Will See
[CLEAR]	Screen Clear	SHIFT	CLR/HOME	
[HOME]	Home		CLR/HOME	
[UP]	Cursor Up	SHIFT	↑CRSR↓	
[DOWN]	Cursor Down		↑CRSR↓	
[LEFT]	Cursor Left	SHIFT	←CRSR→	
[RIGHT]	Cursor Right		←CRSR→	
[SS]	Shifted Space	SHIFT	Space	
[INSERT]	Insert	SHIFT	INST/DEL	
[DEL]	Delete		INST/DEL	
[RVSON]	Reverse On	CNTRL	9	
[RVSOFF]	Reverse Off	CNTRL	0	
[UPARROW]	Up Arrow		↑	
[BACKARROW]	Back Arrow		←	
[PI]	PI		π	
[EP]	English Pound		£	

When You See	It Means	You Type		You Will See
[BLACK]	Black	CNTRL	1	
[WHITE]	White	CNTRL	2	
[RED]	Red	CNTRL	3	
[CYAN]	Cyan	CNTRL	4	
[PURPLE]	Purple	CNTRL	5	
[GREEN]	Green	CNTRL	6	
[BLUE]	Blue	CNTRL	7	
[YELLOW]	Yellow	CNTRL	8	
[F1]	Function 1		F1	
[F2]	Function 2	SHIFT	F1	
[F3]	Function 3		F3	
[F4]	Function 4	SHIFT	F3	
[F5]	Function 5		F5	
[F6]	Function 6	SHIFT	F5	
[F7]	Function 7		F7	
[F8]	Function 8	SHIFT	F7	

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** This page and page 113 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Read these pages **before** entering programs.

BUG REPELLENT

This program will let you debug any *Ahoy!* program. Follow instructions for VIC 20 (cassette or disk) or C-64.

VIC 20 VERSION

By Michael Kleinert and David Barron

For cassette: type in and save the *Bug Repellent* program, then type RUN 63000[RETURN]SYS 828[RETURN]. If you typed the program properly, it will generate a set of two-letter line codes that will match those listed to the right of the respective program lines.

Once you've got a working *Bug Repellent*, type in the program you wish to check. Save it and type the RUN and SYS commands listed above once again, then compare the line codes generated to those listed in the magazine. If you spot a discrepancy, a typing error exists in that line. Important: you must use exactly the same spacing as the program in the magazine. Due to memory limitations on the VIC, the VIC *Bug Repellent* will register an error if your spacing varies from what's printed.

You may type SYS 828 as many times as you wish, but if you use the cassette for anything, type RUN 63000 to restore the *Repellent*.

When your program has been disinfected you may delete all lines from 63000 on. (Be sure the program you type doesn't include lines above 63000!)

For disk: enter *Bug Repellent*, save it, and type RUN:NEW [RETURN]. Type in the program you wish to check, then SYS 828.

To pause the line codes listing, press SHIFT.

To send the list to the printer type OPEN 4,4:CMD 4:SYS 828[RETURN]. When the cursor comes back, type PRINT#4:CLOSE 4[RETURN].

```
•63000 FORX=828TO1023:READY:POKEX,Y:NEXT:END   AC
•63001 DATA169,0,133,63,133,64,165,43,133,251  JL
•63002 DATA165,44,133,252,160,0,132,254,32,228 DF
•63003 DATA3,234,177,251,208,3,76,208,3,230    OE
•63004 DATA251,208,2,230,252,169,244,160,3,32  OH
•63005 DATA30,203,160,0,177,251,170,230,251,20
 8                                             KO
•63006 DATA2,230,252,177,251,32,205,221,169,58 JJ
•63007 DATA32,210,255,169,0,133,253,230,254,32 OK
•63008 DATA228,3,234,165,253,160,0,170,177,251 LG
•63009 DATA201,32,240,6,138,113,251,69,254,170 BP
•63010 DATA138,133,253,177,251,208,226,165,253
 ,41                                           DD
•63011 DATA240,74,74,74,74,24,105,65,32,210    EK
•63012 DATA255,165,253,41,15,24,105,65,32,210  FO
•63013 DATA255,169,13,32,210,255,173,141,2,41  PK
•63014 DATA1,208,249,230,63,208,2,230,64,230   CB
•63015 DATA251,208,2,230,252,76,74,3,169,236   KH
•63016 DATA160,3,32,30,203,166,63,165,64,32    DP
•63017 DATA205,221,169,13,32,210,255,96,230,25
 1                                             EL
•63018 DATA208,2,230,252,96,0,76,73,78,69      OI
•63019 DATA83,58,32,0,76,73,78,69,32,35        FG
•63020 DATA32,0,0,0,0,0                        LE
```

C-64 VERSION

By Michael Kleinert and David Barron

Type in, SAVE, and RUN the *Bug Repellent*. Type NEW, then type in or LOAD the *Ahoy!* program you wish to check. When that's done, SAVE your program (don't RUN it!) and type SYS 49152 [RETURN].

To pause the listing depress and hold the SHIFT key.

Compare the codes your machine generates to the codes listed to the right of the respective program lines. If you spot a difference, an error exists in that line. Jot down the number of lines where contradictions occur. LIST each line, spot the errors, and correct them.

```
•5000 FORX=49152TO49488:READY:POKEX,Y:NEXT:END GJ
•5001 DATA32,161,192,165,43,133,251,165,44,133 DL
•5002 DATA252,160,0,132,254,32,37,193,234,177  DB
•5003 DATA251,208,3,76,138,192,230,251,208,2   OF
•5004 DATA230,252,76,43,192,76,73,78,69,32     KN
•5005 DATA35,32,0,169,35,160,192,32,30,171     CA
•5006 DATA160,0,177,251,170,230,251,208,2,230  CE
•5007 DATA252,177,251,32,205,189,169,58,32,210 JE
•5008 DATA255,169,0,133,253,230,254,32,37,193  CL
•5009 DATA234,165,253,160,0,76,13,193,133,253  NB
•5010 DATA177,251,208,237,165,253,41,240,74,74 MB
•5011 DATA74,74,24,105,65,32,210,255,165,253   EP
•5012 DATA41,15,24,105,65,32,210,255,169,13    GH
•5013 DATA32,220,192,230,63,208,2,230,64,230   AN
•5014 DATA251,208,2,230,252,76,11,192,169,153  NG
•5015 DATA160,192,32,30,171,166,63,165,64,76   BF'
•5016 DATA231,192,96,76,73,78,69,83,58,32      EP
•5017 DATA0,169,247,160,192,32,30,171,169,3    PJ
•5018 DATA133,254,32,228,255,201,83,240,6,201  FK
•5019 DATA80,208,245,230,254,32,210,255,169,4  FL
•5020 DATA166,254,160,255,32,186,255,169,0,133 CL
•5021 DATA63,133,64,133,2,32,189,255,32,192    GC
•5022 DATA255,166,254,32,201,255,76,73,193,96  NN
•5023 DATA32,210,255,173,141,2,41,1,208,249    NH
•5024 DATA96,32,205,189,169,13,32,210,255,32   IM
•5025 DATA204,255,169,4,76,195,255,147,83,67   KC
•5026 DATA82,69,69,78,32,79,82,32,80,82        DC
•5027 DATA73,78,84,69,82,32,63,32,0,76         ML
•5028 DATA44,193,234,177,251,201,32,240,6,138  GN
•5029 DATA113,251,69,254,170,138,76,88,192,0   JK
•5030 DATA0,0,0,230,251,208,2,230,252,96       NA
•5031 DATA170,177,251,201,34,208,6,165,2,73    DM
•5032 DATA255,133,2,165,2,208,218,177,251,201  JA
•5033 DATA32,208,212,198,254,76,29,193,0,169   FM
•5034 DATA13,76,210,255,0,0,0                  PA
```

FLANKSPEED FOR THE C-64

By Gordon F. Wheat

Flankspeed will allow you to enter machine language *Ahoy!* programs without any mistakes. Once you have typed the program in, save it for future use. While entering an ML program with *Flankspeed* there is no need to enter spaces or hit the carriage return. This is all done automatically. If you make an error in a line a bell will ring and you will be asked to enter it again. To LOAD in a program Saved with *Flankspeed* use LOAD "name",1,1 for tape, or LOAD "name",8,1 for disk. The function keys may be used after the starting and ending addresses have been entered.

f1—SAVEs what you have entered so far.

f3—LOADs in a program worked on previously.

f5—To continue on a line you stopped on after LOADing in the previously saved work.

f7—Scans through the program to locate a particular line, or to find out where you stopped the last time you entered the program. f7 temporarily freezes the output as well.

```
•5 POKE53280,12:POKE53281,11                   LL
•6 PRINT"[CLEAR][c 8][RVSON][15" "]FLANKSPEED[
 15" "]";                                      ED
•10 PRINT"[RVSON][5" "]MISTAKEPROOF ML ENTRY P
 ROGRAM[6" "]"                                 MC
•15 PRINT"[RVSON][9" "]CREATED BY G. F. WHEAT[
 9" "]"                                        DM
•20 PRINT"[RVSON][3" "]COPR. 1984, ION INTERNA
```

```
TIONAL INC.[3" "]"                               DH
·30 FORA=54272TO54296:POKEA,0:NEXT               IM
·40 POKE54272,4:POKE54273,48:POKE54277,0:POKE5
4278,249:POKE54296,15                            NH
·70 FORA=680TO699:READB:POKEA,B:NEXT             KO
·75 DATA169,251,166,253,164,254,32,216,255,96    HJ
·76 DATA169,0,166,251,164,252,32,213,255,96      JB
·80 B$="STARTING ADDRESS IN HEX":GOSUB2010:AD=
B:SR=B                                           HC
·85 GOSUB2520:IFB=0THEN80                        FO
·86 POKE251,T(4)+T(3)*16:POKE252,T(2)+T(1)*16    KE
·90 B$="ENDING ADDRESS IN HEX":GOSUB2010:EN=B    IF
·95 GOSUB2510:IFB=0THEN80                        FP
·96 POKE254,T(2)+T(1)*16:B=T(4)+1+T(3)*16        MN
·97 IFB>255THENB=B-255:POKE254,PEEK(254)+1       GE
·98 POKE253,B:PRINT                              HN
·100 REM  GET HEX LINE                           IL
·110 GOSUB3010:PRINT": [c P][LEFT]";:FORA=0TO8   FG
·120 FORB=0TO1:GOTO210                           MD
·125 NEXTB                                       ME
·130 A%(A)=T(1)+T(0)*16:IFAD+A-1=ENTHEN310       LH
·135 PRINT"[c P][LEFT]";                         IK
·140 NEXTA:T=AD-(INT(AD/256)*256):PRINT" "       PD
·150 FORA=0TO7:T=T+A%(A):IFT>255THENT=T-255      LK
·160 NEXT                                        IA
·170 IFA%(8)<>TTHENGOSUB1010:GOTO110             FK
·180 FORA=0TO7:POKEAD+A,A%(A):NEXT:AD=AD+8:GOT
O110                                             MN
·200 REM    GET HEX INPUT                        AB
·210 GETA$:IFA$=""THEN210                        HO
·211 IFA$=CHR$(20)THEN270                        GC
·212 IFA$=CHR$(133)THEN4000                      MD
·213 IFA$=CHR$(134)THEN4100                      KF
·214 IFA$=CHR$(135)THENPRINT" ":GOTO4500         GE
·215 IFA$=CHR$(136)THENPRINT" ":GOTO4700         BJ
·220 IFA$>"@"ANDA$<"G"THENT(B)=ASC(A$)-55:GOTO
250                                              GM
·230 IFA$>"/"ANDA$<":"THENT(B)=ASC(A$)-48:GOTO
250                                              LE
·240 GOSUB1100:GOTO210                           LL
·250 PRINTA$"[c P][LEFT]";                       OA
·260 GOTO125                                     CG
·270 IFA>0THEN280                                OP
·272 A=-1:IFB=1THEN290                           OB
·274 GOTO140                                     CJ
·280 IFB=0THENPRINTCHR$(20);CHR$(20);:A=A-1      HG
·285 A=A-1                                       BE
·290 PRINTCHR$(20);:GOTO140                      KH
·300 REM LAST LINE                               AD
·310 PRINT" ":T=AD-(INT(AD/256)*256)             GJ
·320 FORB=0TOA-1:T=T+A%(B):IFT>255THENT=T-255    PL
·330 NEXT                                        IA
·340 IFA%(A)<>TTHENGOSUB1010:GOTO110             KF
·350 FORB=0TOA-1:POKEAD+B,A%(B):NEXT             HN
·360 PRINT:PRINT"YOU ARE FINISHED!":GOTO4000     ON
·1000 REM BELL AND ERROR MESSAGES                FL
·1010 PRINT:PRINT"LINE ENTERED INCORRECTLY":PR
INT:GOTO1100                                     DH
·1020 PRINT:PRINT"INPUT A 4 DIGIT HEX VALUE!":
GOTO1100                                         JA
·1030 PRINT:PRINT"ENDING IS LESS THAN STARTING
!":B=0:GOTO1100                                  HD
·1040 PRINT:PRINT"ADDRESS NOT WITHIN SPECIFIED
 RANGE!":B=0:GOTO1100                            AG
·1050 PRINT:PRINT"NOT ZERO PAGE OR ROM!":B=0:G
OTO1100                                          KN
·1060 PRINT"?ERROR IN SAVE":GOTO1100             EI
·1070 PRINT"?ERROR IN LOAD":GOTO1100             GL
·1080 PRINT:PRINT:PRINT"END OF ML AREA":PRINT    PG
·1100 POKE54276,17:POKE54276,16:RETURN           BH
·1200 OPEN15,8,15:INPUT#15,A,A$:CLOSE15:PRINTA
$:RETURN                                         IM
·2000 REM GET FOUR DIGIT HEX                     PC
·2010 PRINT:PRINTB$;:INPUTT$                     GM
·2020 IFLEN(T$)<>4THENGOSUB1020:GOTO2010         II
·2040 FORA=1TO4:A$=MID$(T$,A,1):GOSUB2060:IFT(
A)=16THENGOSUB1020:GOTO2010                      AD
·2050 NEXT:B=(T(1)*4096)+(T(2)*256)+(T(3)*16)+
T(4):RETURN                                      GF
·2060 IFA$>"@"ANDA$<"G"THENT(A)=ASC(A$)-55:RET
URN                                              EH
·2070 IFA$>"/"ANDA$<":"THENT(A)=ASC(A$)-48:RET
URN                                              KP
·2080 T(A)=16:RETURN                             NP
·2500 REM ADRESS CHECK                           LI
·2510 IFAD>ENTHEN1030                            MI
·2515 IFB<SRORB>ENTHEN1040                       MG
·2520 IFB<256OR(B>40960ANDB<49152)ORB>53247THE
N1050                                            MI
·2530 RETURN                                     IM
·3000 REM ADDRESS TO HEX                         EB
·3010 AC=AD:A=4096:GOSUB3070                     HG
·3020 A=256:GOSUB3070                            CE
·3030 A=16:GOSUB3070                             PN
·3040 A=1:GOSUB3070                              MJ
·3060 RETURN                                     IM
·3070 T=INT(AC/A):IFT>9THENA$=CHR$(T+55):GOTO3
090                                              CJ
·3080 A$=CHR$(T+48)                              JP
·3090 PRINTA$;:AC=AC-A*T:RETURN                  AC
·4000 A$="**SAVE**":GOSUB4200                    AI
·4050 OPEN1,T,1,A$:SYS680:CLOSE1                 LH
·4060 IFST=0THENEND                              EO
·4070 GOSUB1060:IFT=8THENGOSUB1200               FJ
·4080 GOTO4000                                   FF
·4100 A$="**LOAD**":GOSUB4200                    AB
·4150 OPEN1,T,0,A$:SYS690:CLOSE1                 MF
·4160 IFST=64THEN110                             JH
·4170 GOSUB1070:IFT=8THENGOSUB1200               CM
·4180 GOTO4100                                   FO
·4200 PRINT" ":PRINTTAB(14)A$                    FG
·4210 PRINT:A$="":INPUT"FILENAME";A$             OM
·4215 IFA$=""THEN4210                            GF
·4220 PRINT:PRINT"TAPE OR DISK?":PRINT           DF
·4230 GETB$:T=1:IFB$="D"THENT=8:A$="@0:"+A$:RE
TURN                                             IG
·4240 IFB$<>"T"THEN4230                          FN
·4250 RETURN                                     IM
·4500 B$="CONTINUE FROM ADDRESS":GOSUB2010:AD=
B                                                DK
·4510 GOSUB2515:IFB=0THEN4500                    MA
·4520 PRINT:GOTO110                              OI
·4700 B$="BEGIN SCAN AT ADDRESS":GOSUB2010:AD=
B                                                FH
·4705 GOSUB2515:IFB=0THEN4700                    NK
·4706 PRINT:GOTO4740                             DI
·4710 FORB=0TO7:AC=PEEK(AD+B):GOSUB3030:IFAD+B
=ENTHENAD=SR:GOSUB1080:GOTO110                   BK
·4715 PRINT" ";:NEXTB                            EC
·4720 PRINT:AD=AD+8                              GN
·4730 GETB$:IFB$=CHR$(136)THEN110                MN
·4740 GOSUB3010:PRINT": ";:GOTO4710              JD
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

COMPUPHOBIA
FROM PAGE 77

SLIPSHOD VERSION

```
·10 PRINT"SLIPSHOD VERSION"                    PP
·20 PRINT" COPYRIGHT 1985 CHERYL PETERSON
 "                                             PA
·130 PRINT"USING MY MICROWAVE'S AUTOMATIC
 TIMER AND PRE-PROGRAMMED"                     NB
·131 PRINT"COOKING CYCLES DISTURBS ME"         DN
·132 PRINT"1) NONE"                            JK
·133 PRINT"2) A LITTLE"                        KN
·134 PRINT"3) SOME"                            IG
·135 INPUT"4) VERY MUCH";A                     JB
·136 IF A=1THENT=0                             JP
·137 IF A=2THENT=1                             JF
·138 IF A=3THENT=2                             KD
·139 IF A=4THENT=3                             JJ
·148 IF A>0ANDA<5THEN GOTO 230                 FK
·149 PRINT"TRY A NUMBER, 1-4":GOTO 130         EE
·230 PRINT"USING THE AUTOMATIC RECORDING
 FEATURES OF A VCR BOTHERS ME"                 LM
·232 PRINT"1) NONE"                            JK
·233 PRINT"2) A LITTLE"                        KN
·234 PRINT"3) SOME"                            IG
·235 INPUT"4) VERY MUCH";A                     JB
·236 IF A=1THENT=T+0                           KM
·237 IF A=2THENT=T+1                           JE
·238 IF A=3THENT=T+2                           LA
·239 IF A=4THENT=T+3                           II
·248 IF A>0ANDA<5THEN GOTO 330                 FL
·249 PRINT"TRY A NUMBER, 1-4":GOTO 230         EF
·330 PRINT"WATCHING A TELEVISION SET OR M
 ONITOR FOR HOURS"                             AJ
·331 PRINT"1) GIVES ME A HEADACHE, SO I D
 ON'T DO IT."                                  KG
·332 PRINT"2) MAKES MY EYES HURT SO I DON
 'T DO IT."                                    MM
·333 PRINT"3) KEEPS ME PLEASANTLY OCCUPIE
 D."                                           DP
·334 INPUT"4) IS A WAY OF LIFE.";A             CC
·336 IF A=1THENT=T+3                           LB
·337 IF A=2THENT=T+2                           JH
·338 IF A=3THENT=T+1                           LB
·339 IF A=4THENT=T+0                           IH
·348 IF A>0ANDA<5THEN GOTO 430                 FE
·349 PRINT"TRY A NUMBER, 1-4":GOTO 330         EG
·430 PRINT"A REMOTE CONTROL DEVICE FOR A
 TELEVISION IS "                               DG
·431 PRINT"1) A NUISANCE."                     EO
·432 PRINT"2) A WASTE OF TIME."                FL
·433 PRINT"3) SOMETIMES CONVENIENT."           BH
·434 INPUT"4) WONDERFUL.";A                    LN
·436 IF A=1THENT=T+3                           LB
·437 IF A=2THENT=T+2                           JH
·438 IF A=3THENT=T+1                           LB
·439 IF A=4THENT=T+0                           IH
·448 IF A>0ANDA<5THEN GOTO 530                 FF
·449 PRINT"TRY A NUMBER, 1-4":GOTO 430         BH
·530 PRINT"AUTOMATIC TELLER MACHINES ARE"      NC
·531 PRINT"1) A NUISANCE."                     EO
·532 PRINT"2) A WASTE OF TIME."                FL
·533 PRINT"3) SOMETIMES CONVENIENT."           BH
·534 INPUT"4) WONDERFUL.";A                    LN
·536 IF A=1THENT=T+3                           LB
·537 IF A=2THENT=T+2                           JH
·538 IF A=3THENT=T+1                           LB
·539 IF A=4THENT=T+0                           IH
·548 IF A>0ANDA<5THEN GOTO 630                 FG
·549 PRINT"TRY A NUMBER, 1-4":GOTO 530         BA
·630 PRINT"I FIND USING A SELF CORRECTING
 TYPEWRITER"                                   PG
·631 PRINT"1) FRIGHTENING."                    LB
·632 PRINT"2) CONFUSING."                      EG
·633 PRINT"3) BETTER THEN USING A MANUAL.
 "                                             KI
·634 INPUT"4) A ROYAL PAIN COMPARED TO US
 ING A WORD PROCESSOR.";A                      KB
·636 IF A=1THENT=T+3                           LB
·637 IF A=2THENT=T+2                           JH
·638 IF A=3THENT=T+1                           LB
·639 IF A=4THENT=T+0                           IH
·648 IF A>0ANDA<5THEN GOTO 730                 GH
·649 PRINT"TRY A NUMBER, 1-4":GOTO 630         BB
·730 PRINT"SETTING THE AUTO-DIALING FEATU
 RE ON A PHONE IS"                             GH
·731 PRINT"1) BEYOND ME."                      AB
·732 PRINT"2) TOO MUCH BOTHER."                LE
·733 PRINT"3) WORTH IT IN CONVENIENCE."        IA
·734 INPUT"4) EASY, BUT I LET MY COMPUTER
 DO ALL MY DIALING.";A                         IA
·736 IF A=1THENT=T+3                           LB
·737 IF A=2THENT=T+2                           JH
·738 IF A=3THENT=T+1                           LB
·739 IF A=4THENT=T+0                           IH
·748 IF A>0ANDA<5THEN GOTO 830                 GA
·749 PRINT"TRY A NUMBER, 1-4":GOTO 730         BC
·830 PRINT"IF MY CAR'S DASHBOARD LOOKED L
 IKE THE COCKPIT OF THE SPACE SHUTTLE"         PD
·831 PRINT"1) I'D SELL IT"                     CH
·832 PRINT"2) I'D IGNORE IT."                  MP
·833 PRINT"3) EVENTUALLY I'D FIGURE EVERY
 THING OUT."                                   IL
·834 INPUT"4) I'D BE FIGURING OUT HOW TO
 GET COMPUTER MAPPING RUNNING";A               LA
·836 IF A=1THENT=T+3                           LB
·837 IF A=2THENT=T+2                           JH
·838 IF A=3THENT=T+1                           LB
·839 IF A=4THENT=T+0                           IH
·848 IF A>0ANDA<5THEN GOTO 930                 GB
·849 PRINT"TRY A NUMBER, 1-4":GOTO 830         CD
```

```
•930 PRINT"I'D LIKE TO HAVE A COMPUTER BE
CAUSE"                                      LN
•931 PRINT"1) IT'S A GREAT STATUS SYMBOL.
"                                           ND
•932 PRINT"2) MY KIDS WILL NEED ONE."       CN
•933 PRINT"3) THERE MUST BE SOMETHING I C
AN DO WITH ONE."                            OF
•934 INPUT"4) I CAN THINK OF AT LEAST A D
OZEN THINGS TO DO WITH ONE";A               AF
•936 IF A=1THENT=T+3                        LB
•937 IF A=2THENT=T+2                        JH
•938 IF A=3THENT=T+1                        LB
•939 IF A=4THENT=T+0                        IH
•948 IF A>0ANDA<5THEN GOTO 1030             KB
•949 PRINT"TRY A NUMBER, 1-4":GOTO 930      BM
•1030 PRINT"I WOULDN'T GO THROUGH THE RID
ES AT EPCOT CENTER BECAUSE"                 OB
•1031 PRINT"1) I DON'T TRUST THOSE COMPUT
ER RUN CONTRAPTIONS."                       ID
•1032 PRINT"2) WHO CARES ABOUT TECHNO MUM
BLE ANYWAY."                                EE
•1033 PRINT"3) I'D BE TOO BUSY EATING!"     CA
•1034 INPUT"4) I'VE ALREADY RIDDEN ON ALL
 OF THEM";A                                 DJ
•1036 IF A=1THENT=T+3                       LB
•1037 IF A=2THENT=T+2                       JH
•1038 IF A=3THENT=T+1                       LB
•1039 IF A=4THENT=T+0                       IH
•1048 IF A>0ANDA<5THEN GOTO 1100            JP
•1049 PRINT"TRY A NUMBER, 1-4":GOTO 1030    IO
•1100 INPUT "HOW MANY COMPUTERS DO YOU OW
N";A                                        BP
•1105 IF A>10 THEN PRINT"TRY A NUMBER LES
S THAN 11"                                  BK
•1110 IF A>10 THEN GOTO 1100                ML
•1120 IF T>10 OR T=10 THEN T=T-A            NC
•1300 IF T>10 THEN 1400                     HL
•1310 PRINT"YOUR SCORE IS A MEASLY";T;"WH
ICH MEANS YOU ARE AT HOME WITH"             IG
•1320 PRINT"THE COMING COMPUTER REVOLUTIO
N. IF YOU AREN'T ALREADY ACTIVELY"          PB
•1330 PRINT"INVOLVED WITH THE BEASTIES, Y
OU SHOULD BE."                              CO
•1340 END                                   IC
•1400 IF T>20 THEN 1500                     GF
•1405 PRINT"YOUR SCORE IS";T;"."            PC
•1410 PRINT"WITH A VERY SLIGHT ATTITUDE A
DJUSTMENT, YOU COULD BE"                    AA
•1420 PRINT"HAPPILY ENJOYING THE BENEFITS
 OF A COMPUTER COMPANION."                  NI
•1440 END                                   IC
•1500 IF T>25 THEN 1600                     FD
•1505 PRINT"YOUR SCORE IS";T;"."            PC
•1510 PRINT"GETTING YOU INTO COMMUNION WI
TH A COMPUTER WILL TAKE"                    EJ
•1520 PRINT"A TREMENDOUS EFFORT. BETTER O
FF LEAVING TO THE NEXT GENERATION."         OB
•1540 END                                   IC
•1599 PRINT"YOUR SCORE IS";T;"!"            ME
•1600 PRINT"FORGET IT! IT'S HOPELESS. A C
LASSIC CASE OF COMPUPHOBIA."                DP
•1610 PRINT"HOW DID YOU SURVIVE TAKING TH
IS TEST?!"                                  MG
```

STRUCTURED VERSION

```
•10 PRINT"STRUCTURED VERSION"               DN
•20 PRINT"COPYRIGHT 1985 CHERYL PETERSON"   EB
•30 PRINT"HIT ANY KEY TO CONTINUE"          OP
•31 GET Z$:IF Z$=""THEN 31                  FF
•100 PRINT"[CLEAR]"                         HH
•130 PRINT"USING MY MICROWAVE'S AUTOMATIC
 TIMER AND PRE-PROGRAMMED"                  NB
•131 PRINT"COOKING CYCLES DISTURBS ME"      DN
•132 GOSUB 10001                            IL
•145 GOSUB 5000                             FK
•146 IF A%<0ORA%>4 THEN 130                 BG
•150 GOSUB 10100                            IN
•230 PRINT"USING THE AUTOMATIC RECORDING
FEATURES OF A VCR BOTHERS ME"               LM
•232 GOSUB 10001                            IL
•245 GOSUB 5000                             FK
•246 IF A%<0ORA%>4 THEN 230                 BL
•260 GOSUB 10100                            IN
•330 PRINT"WATCHING A TELEVISION SET OR M
ONITOR FOR HOURS"                           AJ
•331 PRINT"1) GIVES ME A HEADACHE, SO I D
ON'T DO IT."                                KG
•332 PRINT"2) MAKES MY EYES HURT, SO I DO
N'T DO IT."                                 FL
•333 PRINT"3) KEEPS ME PLEASANTLY OCCUPIE
D."                                         DP
•334 INPUT"4) IS A WAY OF LIFE.";A%         FP
•340 GOSUB 5000                             FK
•350 IF A%<0ORA%>4 THEN 330                 BI
•360 GOSUB 10200                            IK
•430 PRINT"A REMOTE CONTROL DEVICE FOR A
TELEVISION IS "                             DG
•431 GOSUB 10050                            JJ
•440 GOSUB 5000                             FK
•450 IF A%<0ORA%>4 THEN 430                 BN
•460 GOSUB 10200                            IK
•530 PRINT"AUTOMATIC TELLER MACHINES ARE"   NC
•531 GOSUB 10050                            JJ
•540 GOSUB 5000                             FK
•550 IF A%<0ORA%>4 THEN 530                 BK
•560 GOSUB 10200                            IK
•630 PRINT"I FIND USING A SELF CORRECTING
 TYPEWRITER"                                PG
•631 PRINT"1) FRIGHTENING."                 LB
•632 PRINT"2) CONFUSING."                   EG
•633 PRINT"3) BETTER THAN USING A MANUAL.
"                                           HM
•634 INPUT"4) A ROYAL PAIN COMPARED TO US
ING A WORD PROCESSOR.";A%                   LN
•640 GOSUB 5000                             FK
•650 IF A%<0ORA%>4 THEN 630                 DP
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
·660 GOSUB 10200                                    IK
·730 PRINT"SETTING THE AUTO-DIALING FEATU
 RE ON A PHONE IS"                                  GH
·731 PRINT"1) BEYOND ME."                           AB
·732 PRINT"2) TOO MUCH BOTHER."                     LE
·733 PRINT"3) WORTH IT IN CONVENIENCE."             IA
·734 INPUT"4) EASY, BUT I LET MY COMPUTER
  DO ALL MY DIALING.";A%                            NC
·740 GOSUB 5000                                     FK
·750 IF A%<0ORA%>4 THEN 730                         DM
·760 GOSUB 10200                                    IK
·830 PRINT"IF MY CAR'S DASHBOARD LOOKED L
 IKE THE COCKPIT OF THE SPACE SHUTTLE"              PD
·831 PRINT"1) I'D SELL IT"                          CH
·832 PRINT"2) I'D IGNORE IT."                       MP
·833 PRINT"3) EVENTUALLY I'D FIGURE EVERY
 THING OUT."                                        IL
·834 INPUT"4) I'D BE FIGURING OUT HOW TO
 GET COMPUTER MAPPING SYSTEM RUNNING";A%            MF
·840 GOSUB 5000                                     FK
·850 IF A%<0ORA%>4 THEN 830                         EB
·860 GOSUB 10200                                    IK
·930 PRINT"I'D LIKE TO HAVE A COMPUTER BE
 CAUSE"                                             LN
·931 PRINT"1) IT'S A GREAT STATUS SYMBOL.
 "                                                  ND
·932 PRINT"2) MY KIDS WILL NEED ONE."               CN
·933 PRINT"3) THERE MUST BE SOMETHING I C
 AN DO WITH ONE."                                   OF
·934 INPUT"4) I CAN THINK OF AT LEAST A D
 OZEN THINGS TO DO WITH ONE";A%                     GA
·940 GOSUB 5000                                     FK
·950 IF A%<0ORA%>4 THEN 930                         DO
·960 GOSUB 10200                                    IK
·1030 PRINT"I WOULDN'T GO THROUGH THE RID
 ES AT EPCOT CENTER BECAUSE"                        OB
·1031 PRINT"1) I DON'T TRUST THOSE COMPUT
 ER RUN CONTRAPTIONS."                              ID
·1032 PRINT"2) WHO CARES ABOUT ALL THAT T
 ECHNO-MUMBLE ANYWAY."                              KE
·1033 PRINT"3) I'D BE TOO BUSY EATING!"             CA
·1034 INPUT"4) I'VE ALREADY RIDDEN ON ALL
  OF THEM";A%                                       DP
·1040 GOSUB 5000                                    FK
·1050 IF A%<0ORA%>4 THEN 1030                       FH
·1060 GOSUB 10200                                   IK
·1100 INPUT "HOW MANY COMPUTERS DO YOU OW
 N";A%                                              GF
·1105 IF A%>10 THEN PRINT"TRY A NUMBER LE
 SS THAN 11"                                        CE
·1110 IF A%>10 THEN GOTO 1100                       AE
·1120 IF T>10 OR T=10 THEN T=T-A%                   OK
·1300 IF T>10 THEN 1400                             HL
·1310 PRINT"YOUR SCORE IS A MEASLY";T;"WH
 ICH MEANS YOU ARE AT HOME WITH"                    IG
·1320 PRINT"THE COMING COMPUTER REVOLUTIO
 N. IF YOU AREN'T ALREADY ACTIVELY"                 PB
·1330 PRINT"INVOLVED WITH THE BEASTIES, Y
 OU SHOULD BE."                                     CO
·1340 END                                           IC
·1400 IF T>20 THEN 1500                             GF
·1405 PRINT"YOUR SCORE IS";T;"."                    PC
·1410 PRINT"WITH A VERY SLIGHT ATTITUDE A
 DJUSTMENT, YOU COULD BE"                           AA
·1420 PRINT"HAPPILY ENJOYING THE BENEFITS
  OF A COMPUTER COMPANION."                         NI
·1440 END                                           IC
·1500 IF T>25 THEN 1599                             HC
·1505 PRINT"YOUR SCORE IS";T;"."                    PC
·1510 PRINT"GETTING YOU INTO COMMUNION WI
 TH A COMPUTER WILL TAKE A"                         LC
·1520 PRINT"MIRACLE. BUT STRANGER THINGS
 HAVE HAPPENED. GOOD LUCK!"                         HM
·1540 END                                           IC
·1599 PRINT"YOUR SCORE IS";T;"!"                    ME
·1600 PRINT"FORGET IT! IT'S HOPELESS. A C
 LASSIC CASE OF COMPUPHOBIA."                       DP
·1610 PRINT"HOW DID YOU SURVIVE TAKING TH
 IS TEST?!"                                         MG
·2000 END                                           IC
·5000 IF A%>0ANDA%<5THEN RETURN                     CJ
·5005 PRINT"[CLEAR]"                                HH
·5010 PRINT"TRY CHOOSING AN INTEGER BETWE
 EN 1 AND 4"                                        NE
·10001 PRINT"1) NONE"                               JK
·10002 PRINT"2) A LITTLE"                           KN
·10003 PRINT"3) SOME"                               IG
·10004 INPUT"4) VERY MUCH";A%                       KI
·10005 RETURN                                       IM
·10050 PRINT"1) A NUISANCE."                        EO
·10055 PRINT"2) A WASTE OF TIME."                   FL
·10060 PRINT"3) SOMETIMES CONVENIENT."              BH
·10065 INPUT"4) WONDERFUL.";A%                      ND
·10070 RETURN                                       IM
·10100 IF A%=1THENT=T+0                             MI
·10110 IF A%=2THENT=T+1                             MM
·10120 IF A%=3THENT=T+2                             MI
·10130 IF A%=4THENT=T+3                             KM
·10140 PRINT"[CLEAR]"                               HH
·10150 RETURN                                       IM
·10200 IF A%=1THENT=T+3                             NF
·10210 IF A%=2THENT=T+2                             ML
·10220 IF A%=3THENT=T+1                             MJ
·10230 IF A%=4THENT=T+0                             KP
·10240 PRINT"[CLEAR]"                               HH
·10250 RETURN                                       IM
```

LITTLE WINDOWS
FROM PAGE 18

EXTENDED BACKGROUND

```
·1 REM *** EXTENDED BACKGROUND DEMO ***             BP
```

```
•9 REM SET EXTENDED BACKGROUND COLOR MODE CB
•10 POKE 53265,PEEK(53265)OR 64 AM
•11 REM SET UP REGULAR STRING FOR BACKGRO
 UND #0 PH
•12 A$(0)="ABCDEFGHIJKLMNOPQRSTUVWXYZ 123
 4567890 !#$%&'()+-[EP][BACKARROW]*[UPARR
 OW]@=;:[]/.,<>?" EI
•13 A$(0)=A$(0)+CHR$(34)+CHR$(34) LJ
•14 REM CONVERT REGULAR CHARACTERS TO SHI
 FTED CHARACTERS FOR BACKGROUND #1 OC
•15 A$(1)="":FOR I=1 TO LEN(A$(0)):A=ASC(
 MID$(A$(0),I,1)) FM
•16 IF A<64 THEN A=A+128:GOTO 18 DJ
•17 A=A+32 FB
•18 A$(1)=A$(1)+CHR$(A):NEXT DD
•19 REM ADD REVERSE-ON/OFF CHARACTERS TO
 BOTH STRINGS FOR BACKGROUNDS 2 & 3 CJ
•20 A$(2)="[RVSON]"+A$(0)+"[RVSOFF]":A$(3
 )="[RVSON]"+A$(1)+"[RVSOFF]" JD
•21 FOR I=0 TO 3:PRINT A$(I):NEXT BA
```

WINDOWS

```
•1 REM *** WINDOWS *** NI
•2 REM CREATES THE ILLUSION OF WINDOWS US
 ING EXTENDED BACKGROUND COLOR MODE PA
•13 REM SET EXTENDED COLOR MODE GO
•14 POKE 53265,PEEK(53265)OR 64 AM
•15 GOSUB 800 CI
•19 REM EMPTY THE KEYBOARD BUFFER PP
•20 POKE 198,0 KB
•30 GOTO 100 CF
•39 REM GET RAW JOYSTICK VALUE AND INVERT
  IT (JOYSTICK PORT 2) AA
•40 JY=255-PEEK(56320) LE
•41 REM GET SEPARATE BUTTON VALUE CG
•42 JB=JY AND 16 PK
•43 REM CLEAN UP JOYSTICK VALUE AND RETUR
 N EJ
•44 JY=JY AND 15:RETURN OP
•49 REM ** READ KEYBOARD HH
•50 D=0:GET A$:IF A$="" THEN RETURN LI
•51 REM IS IT A COMMAND CHARACTER? PP
•52 A=ASC(A$):IF A<32 OR A>95 THEN D=A:A=
 0:A$="":RETURN GD
•53 REM CONVERT KEYSTROKE TO CODE FOR CH$
 () ARRAY HK
•54 A=A-32:RETURN BD
•79 REM ** CLEAR WINDOW CK
•80 FOR I=MT(QQ)-1 TO MB(QQ)+1:PRINT VV$(
 I)TAB(ML(QQ)-1); ME
•81 REM SET FOR NON-REVERSE OR REVERSE LK
•82 PRINT "[RVSOFF]";:IF QQ>1 THEN PRINT
 "[RVSON]"; LL
•83 REM PRINT CORRECT PORTION OF WINDOW'S
  BLANK STRING KL
•84 PRINT LEFT$(BL$(QQ),MR(QQ)-ML(QQ)+2);
 :NEXT:RETURN AE
•99 REM ** MAIN KEYREAD LOOP IL
•100 GOSUB 40:GOSUB 50:IF A$="" AND D=0 A
 ND JY=0 AND JB=0 THEN 100 LE
•104 REM IF IT WAS A FUNCTION KEY, CHANGE
  WINDOWS BJ
•105 IF(D>132)AND(D<137)THEN QQ=D-133:POK
 E 53280,BC(QQ):GOTO 100 LA
•109 REM IF IT WAS FUNCTION 2, QUIT FH
•110 IF D=137 THEN SYS 58235 HG
•114 REM IF IT WAS CLR KEY, CLEAR WINDOW BD
•115 IF D=147 THEN HP(QQ)=ML(QQ):VP(QQ)=M
 T(QQ):GOSUB 80:GOTO 100 JK
•119 REM IF IT WAS HOME KEY, HOME CURSOR MD
•120 IF D=19 THEN HP(QQ)=ML(QQ):VP(QQ)=MT
 (QQ):GOTO 100 MG
•124 REM DOWN CURSOR KM
•125 IF D=17 THEN GOSUB 200:GOTO 100 BG
•129 REM UP CURSOR CM
•130 IF D=145 THEN GOSUB 210:GOTO 100 FH
•134 REM RIGHT CURSOR BO
•135 IF D=29 THEN GOSUB 220:GOTO 100 AD
•139 REM LEFT CURSOR LD
•140 IF D=157 THEN GOSUB 230:GOTO 100 FK
•144 REM RETURN KEY GE
•145 IF D=13 OR D=10 THEN GOSUB 200:HP(QQ
 )=ML(QQ):GOTO 100 GJ
•149 REM DELETE KEY DJ
•150 IF D=20 THEN GOSUB 240:GOTO 100 DM
•164 REM IF IT WAS A COMMAND, BUT NOT A L
 EGAL ONE, GO BACK FOR ANOTHER KEYREAD HP
•165 IF D>0 THEN 100 ON
•169 REM IF IT WAS A CHARACTER, PRINT IT HM
•170 PRINT VV$(VP(QQ))TAB(HP(QQ))FC$(QQ)C
 H$(QQ,A);:GOSUB 220:GOTO 100 IA
•190 GOTO 100 CF
•199 REM CURSOR DOWN ROUTINE LD
•200 VP(QQ)=VP(QQ)+1:IF VP(QQ)>MB(QQ) THE
 N VP(QQ)=MT(QQ) NN
•201 RETURN IM
•209 REM CURSOR UP ROUTINE CH
•210 VP(QQ)=VP(QQ)-1:IF VP(QQ)<MT(QQ) THE
 N VP(QQ)=MB(QQ) EI
•211 RETURN IM
•219 REM CURSOR RIGHT ROUTINE AC
•220 HP(QQ)=HP(QQ)+1:IF HP(QQ)>MR(QQ) THE
 N HP(QQ)=ML(QQ):GOSUB 200 AP
•221 RETURN IM
•229 REM CURSOR LEFT ROUTINE PI
•230 HP(QQ)=HP(QQ)-1:IF HP(QQ)<ML(QQ) THE
 N HP(QQ)=MR(QQ):GOSUB 210 AB
•231 RETURN IM
•239 REM DELETE KEY ROUTINE FF
•240 GOSUB 230:PRINT VV$(VP(QQ))TAB(HP(QQ
 ))FC$(QQ)CH$(QQ,0); CP
•241 RETURN IM
•800 DIM VV$(24),CH$(3,63),FC$(3),BC(3),V
 P(3),HP(3),CB(3),MT(3),MB(3),ML(3),MR(3) CK
•801 REM INITIAL COLOR VALUES NF
•802 FOR I=0 TO 3:READ FC$(I),BC(I),CB(I)
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
 :NEXT                                         PG
•803 DATA "[c 8][RVSOFF]",2,53281,"[BLUE]
 [RVSOFF]",14,53282,"[c 2][RVSON]",7,5328
 3,"[YELLOW][RVSON]",5 ,53284                  BI
•804 REM SET MARGINS & STARTING CURSOR PO
 SITIONS                                       LJ
•805 FOR I=0 TO 3:READ MT(I),MB(I),ML(I),
 MR(I):VP(I)=MT(I):HP(I)=ML(I):NEXT            JC
•806 DATA 1,15,1,18,1,9,21,38,18,23,1,18,
 12,23,21,38                                   EJ
•809 REM VERTICAL PLACEMENT STRINGS            CE
•810 VV$(0)="[HOME]":FOR I=1 TO 24:VV$(I)
 =VV$(I-1)+"[DOWN]":NEXT                       PB
•819 REM SET UNSHIFTED CHARACTER TABLES        LG
•820 FOR I=0 TO 63:CH$(0,I)=CHR$(I+32):CH
 $(2,I)=CH$(0,I):NEXT                          GH
•821 REM SET SHIFTED CHARACTER TABLE           IJ
•822 FOR I=0 TO 31:CH$(1,I)=CHR$(I+160):C
 H$(3,I)=CH$(1,I):NEXT                         EG
•823 FOR I=32 TO 63:CH$(1,I)=CHR$(I+64):C
 H$(3,I)=CH$(1,I):NEXT                         NF
•824 REM SET UNSHIFTED BLANK LINE              IH
•825 BL$(0)="[38" "]":BL$(2)=BL$(0)            PF
•826 REM SET SHIFTED BLANK LINE                CP
•827 BL$(1)="":FOR I=1 TO 39:BL$(1)=BL$(1
 )+CHR$(160):NEXT:BL$(3)=BL$(1)                EK
•829 REM INITIALIZE SCREEN                     ID
•830 PRINT "[CLEAR]";:FOR QQ=0 TO 3:GOSUB
  80                                           PA
•831 REM CLEAR WINDOW EDGES                    EA
•832 FOR I=MT(QQ)-1 TO MB(QQ)+1:POKE 1024
 +I*40+MR(QQ)+1,32+64*QQ:NEXT:NEXT             MG
•833 QQ=0                                      IL
•834 FOR I=0 TO 3:POKE CB(I),BC(I):NEXT        JE
•835 POKE 53280,BC(QQ)                         GL
•890 RETURN                                    IM
```

MOVECHRS
FROM PAGE 73

Use of an assembler is required for entry of this program. See introductory article.

```
*
* MOVECHAR
*
            ORG     $8000
*
R6510       EQU     $0001
NEWADR      EQU     $3000
CHRBAS      EQU     $D000
CIACRE      EQU     $DC0E
C2DDRA      EQU     $DD02
MEMREG      EQU     $D018
CHROUT      EQU     $FFD2
*
FRETOP      EQU     $0034
MEMSIZ      EQU     $0038
*
TABLEN      EQU     512
MVSRCE      EQU     $FB
MVDEST      EQU     MVSRCE+2
*
LENPTR      EQU     $200
*
* SET CIA BITS TO OUTPUTS
*
            LDA     C2DDRA
            ORA     #3
            STA     C2DDRA
*
* SWITCH TO UPPER CASE
*
            LDA     #142
            JSR     CHROUT
*
* CLEAR RAM FOR CHR MEMORY
*
            LDA     #48
            STA     FRETOP
            STA     MEMSIZ
*
* TURN OFF KB INTERRUPT TIMER
*
            LDA     CIACRE
            AND     #$FE
            STA     CIACRE
*
* SWITCH I/O OFF, CHAR ROM ON
*
            LDA     R6510
            AND     #$FB
            STA     R6510
*
* COPY CHARACTERS INTO RAM
*
            LDA     #<CHRBAS
            STA     MVSRCE
            LDA     #>CHRBAS
            STA     MVSRCE+1
*
            LDA     #<NEWADR
            STA     MVDEST
            LDA     #>NEWADR
            STA     MVDEST+1
*
            LDA     #<TABLEN
            STA     LENPTR
            LDA     #>TABLEN
            STA     LENPTR+1
*
* START MOVE
*
            LDY     #0
```

```
             LDX   LENPTR+1
             BEQ   MVPART
    MVPAGE   LDA   (MVSRCE),Y
             STA   (MVDEST),Y
             INY
             BNE   MVPAGE
             INC   MVSRCE+1
             INC   MVDEST+1
             DEX
             BNE   MVPAGE
    MVPART   LDX   LENPTR
             BEQ   MVEXIT
    MVLAST   LDA   (MVSRCE),Y
             STA   (MVDEST),Y
             INY
             DEX
             BNE   MVLAST
    MVEXIT
    *
    * SWITCH I/O BACK IN
    *
             LDA   R6510
             ORA   #4
             STA   R6510
    *
    * TURN TIMER BACK ON
    *
             LDA   CIACRE
             ORA   #1
             STA   CIACRE
    *
    * SET VIC MEMORY CONTROL REGISTER
    *
             CLC
             LDA   MEMREG
             AND   #$F0
             ADC   #$0C
             STA   MEMREG
    *
             RTS
    *
             END
    *
```

ELMAC

FROM PAGE 34

ELMAC PLAYS NIM

```
·10 REM                                        JD
·15 REM  RUPERT REPORT #26                     OP
·20 REM   ELMAC PLAYS NIM                      GA
·25 REM                                        JD
·30 PRINT CHR$(147);                           JL
·40 :REM - - - - - - - -                       CL
·50 :REM   INITIALIZE                          NC
·55 :REM - - - - - - - -                       CL
·60 DIM A(3,4,5,12),M(6,3)                     NP
·70 FALSE=0 : TRUE=NOT FALSE                   MH
·80 BK$="[10"[LEFT]"]"                         JC
·90 SP$="[38" "]"                              PK
·100 L$="[38"="]"                              AB
·110 PRINT "ELMAC <THE ELEMENTARY LEARNIN
 G MACHINE>"TAB(53)"PLAYS NIM" : PRINT     NP
·120 OPEN 15,8,15 :REM DISK ERROR CHANNEL NK
·125 :REM - - - - - - - - - - - - -        MP
·130 REM - - BEGIN NEW GAME - -            NC
·140 :REM - - - - - - - - - - - - -        MP
·150 :REM  READ KEYBOARD                   BE
·160 GOSUB 2110                            FL
·170 :REM  PLAYER CHOSEN YET?              EP
·180 IF PL=0 THEN 160                      EH
·190 :REM  MOVE ARRAY FILLED YET?          CG
·200 IF F0=1 THEN 230                      CA
·210 :REM  FILL MOVE ARRAY INITIALLY       AC
·220 PRINT : PRINT "THINKING [3"."]" : GO
 SUB 1580 : F0=1                           NA
·230 IF NOT MDCHG THEN 300                 ML
·240 :REM  NEW GAME IN NEW MODE            KB
·250 PRINT CHR$(147); : LL=5 : GOSUB 2400
 : MDCHG=FALSE                             IB
·260 :REM                                  ML
·270 :REM <><><> MAIN GAME LOOP <><><><>   HL
·280 :REM                                  ML
·290 :REM  NEW GAME IN PREVIOUS MODE       MF
·300 X=3 : Y=4 : Z=5 : MN=0                ON
·310 :REM  DOES ELMAC GO FIRST?            LE
·320 IF RND(0)<.5 THEN 380                 BA
·330 :REM  OTHER PLAYER'S MOVE             PI
·340 ON PL GOSUB 810,1790                  EH
·350 :REM  DID OTHER PLAYER WIN?           FJ
·360 IF X+Y+Z=0 THEN WIN=FALSE :GOTO 460  DK
·370 :REM  ELMAC'S TURN                    OK
·380 GOSUB 530                             CI
·390 :REM  DID ELMAC LOSE?                 JP
·400 IF TM=0 THEN WIN=FALSE : GOTO 460    PK
·410 :REM  DID ELMAC MAKE LAST MOVE?       DL
·420 IF X+Y+Z=0 THEN WIN=TRUE : GOTO 460  MG
·430 :REM  READY FOR NEXT MOVE             HD
·440 GOTO 340                              CD
·450 :REM  GO TO GAME OVER ROUTINE         EK
·460 GOSUB 970                             DA
·470 :REM  HAS ANY KEY BEEN PRESSED?       AA
·480 GET R$ : IF R$<>"" THEN 160           FP
·490 :REM  START NEXT GAME, SAME MODE      AI
·500 GOTO 300                              BP
·510 :REM <><><> END OF GAME LOOP <><><>   IE
·520 :REM - - - - - - - - - -              HC
·530 REM    ELMAC'S MOVE                   JP
·540 :REM - - - - - - - - - -              HC
·550 :REM  HOW MANY VALID MOVES?           EO
·560 TM=A(X,Y,Z,0)                         KA
·570 :REM RETURN IF NO VALID MOVES; ELMAC
 LOST                                      OF
·580 IF TM=0 THEN 790                      FH
·590 :REM  SELECT NEXT MOVE                CO
·600 RN=INT(RND(0)*TM)+1                   OH
·610 :REM  STEP THRU THE CHOICES TO FIND
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
      THE SELECTED MOVE                        DK
•620 CT=0 : FOR P=1 TO 12                      ID
•630 CT=CT+A(X,Y,Z,P) : MV=P                   MG
•640 :REM  FOUND MOVE VALUE (MV=1 TO 12)       BJ
•650 IF CT>=RN THEN P=12                       EA
•660 NEXT P                                    NC
•670 :REM  RECORD THE MOVE                     JD
•680 MN=MN+1 : M(MN,0)=MV : M(MN,1)=X          GC
•690 M(MN,2)=Y : M(MN,3)=Z                     NF
•700 :REM  SAVE THE BOARD                      FB
•710 X0=X : Y0=Y : Z0=Z                        CK
•720 :REM  MV STORES THE SELECTED MOVE         IP
•730 IF MV<4 THEN X=X-MV : GOTO 770            OJ
•740 IF MV<8 THEN Y=Y-MV+3 : GOTO 770          JJ
•750 Z=Z-MV+7                                  JM
•760 :REM  SHOW BOARD AFTER MOVE               PE
•770 PRINT "ELMAC: ";X;Y;Z                     GJ
•780 :REM  NEXT MOVE                           DF
•790 RETURN                                    IM
•800 :REM - - - - - - - - - - - -              KO
•810 REM       RANDOM PLAYER                   AH
•820 :REM - - - - - - - - - - - -              KO
•830 :REM  PICK RANDOM MOVE                    NN
•840 ROW=INT(RND(0)*3)+1                       KD
•850 IF ROW=1 THEN MAX=X                       BG
•860 IF ROW=2 THEN MAX=Y                       AG
•870 IF ROW=3 THEN MAX=Z                       AC
•880 :REM IF ROW EMPTY, TRY AGAIN              KI
•890 IF MAX=0 THEN 840                         KE
•900 NUM=INT(RND(0)*MAX)+1                     HP
•910 IF ROW=1 THEN X=X-NUM                     EM
•920 IF ROW=2 THEN Y=Y-NUM                     CH
•930 IF ROW=3 THEN Z=Z-NUM                     CK
•940 PRINT "RANDOM:";X;Y;Z                     FN
•950 RETURN                                    IM
•960 :REM - - - - - - - - - - -                HC
•970 REM       GAME OVER                       OH
•980 :REM - - - - - - - - - - -                HC
•990 :REM  INCREMENT NUMBER OF GAMES           FO
•1000 NG=NG+1                                  LK
•1010 IF WIN=TRUE THEN 1110                    BB
•1020 :REM  > > ELMAC LOST! < <                IH
•1030 :REM  PUNISH THE LAST MOVE               EH
•1040 LAST=A(X0,Y0,Z0,MV)                      GP
•1050 A(X0,Y0,Z0,MV)=0                         ED
•1060 A(X0,Y0,Z0,0)=A(X0,Y0,Z0,0)-LAST         KI
•1070 :REM  INCREMENT NUMBER OF LOSSES         JL
•1080 NL=NL+1                                  LC
•1090 :REM  UPDATE SCREEN                      FM
•1100 GOTO 1220                                FC
•1110 :REM > > ELMAC WON! < <                  EL
•1120 :REM  INCREMENT NUMBER OF WINS           CM
•1130 NW=NW+1                                  NK
•1140 :REM  REWARD ALL WINNING MOVES           PO
•1150 FOR J=1 TO MN                            OC
•1160 X1=M(J,1) : Y1=M(J,2) : Z1=M(J,3)        CG
•1170 M1=M(J,0)                                JE
•1180 A(X1,Y1,Z1,M1)=A(X1,Y1,Z1,M1)+1          OA
•1190 A(X1,Y1,Z1,0)=A(X1,Y1,Z1,0)+1            OF
•1200 NEXT J                                   MM
•1210 :REM  UPDATE SCREEN                      FM
•1220 IF PL=1 THEN 1260                        HO
•1230 IF WIN THEN PRINT "ELMAC WON. PRESS
 ANY KEY."                                     KL
•1240 IF NOT WIN THEN PRINT "YOU WON. PRE
SS ANY KEY."                                   ED
•1250 GET R$ : IF R$="" THEN 1250              LG
•1260 FOR J=1 TO 8 : PRINT SP$ : NEXT J :
LL=1 : GOSUB 2400                              DI
•1270 WSTRK=WSTRK+1                            MN
•1280 IF NOT WIN THEN WSTRK=0                  FN
•1290 IF WSTRK>WMAX THEN WMAX=WSTRK            DB
•1300 PRINT "# GAMES, WINS, LOSSES[4" "]%
 WIN"                                          PE
•1310 PRINT TAB(2)NG;TAB(9)NW;TAB(16)NL;T
AB(25)LEFT$(STR$(100*NW/NG)+"[4" "]",7)        MN
•1320 PRINT "CURRENT/LONGEST WIN STREAK[1
0" "]";BK$;WSTRK;"/";WMAX : PRINT L$           KI
•1330 :REM  READY FOR NEW GAME                 AJ
•1340 RETURN                                   IM
•1350 :REM - - - - - - - - - - - - - - - -     EO
•1360 REM  DISPLAY & SAVE MOVE ARRAY           PA
•1370 :REM - - - - - - - - - - - - - - - -     EO
•1375 :REM  DISPLAY MOVE ARRAY                 HP
•1380 PRINT CHR$(147)                          FG
•1390 FOR I=0 TO 3 : FOR J=0 TO 4              DI
•1400 FOR K=0 TO 5 : PRINT "[RVSON]";I;J;
K;"[RVSOFF]"                                   EJ
•1410 FOR P=0 TO 12 : PRINT A(I,J,K,P); :
 NEXT P                                        EB
•1420 PRINT :GET R$ :IF R$<>"" THEN I=3 :
J=4 : K=5                                      NA
•1430 NEXT K : NEXT J : NEXT I                 MP
•1440 :REM  SAVE MOVE ARRAY TO DISK            CM
•1450 INPUT "SAVE THIS ARRAY (Y/N)";A$         NG
•1460 IF A$<>"Y" THEN 1560                     IK
•1470 INPUT "WHAT FILENAME";F$                 DL
•1480 :REM  CREATE MOVE ARRAY IF IT'S NOT
 ALREADY LOADED                                MM
•1490 IF F0=0 THEN PRINT "ONE MOMENT [3".
"]" : GOSUB 1580 : F0=1                        FL
•1500 OPEN 1,8,8,F$+",S,W" : GOSUB 2500        AO
•1510 FOR I=0 TO 3 : FOR J=0 TO 4              DI
•1520 FOR K=0 TO 5 : FOR P=0 TO 12             ID
•1530 PRINT#1,A(I,J,K,P)                       DP
•1540 NEXT P,K,J,I                             EJ
•1550 PRINT#1 : CLOSE 1                        NE
•1560 RETURN                                   IM
•1570 :REM - - - - - - - - - - - - - - - -     EO
•1580 REM  FILL MOVE ARRAY INITIALLY           NC
•1590 :REM - - - - - - - - - - - - - - - -     EO
•1600 FOR X=0 TO 3 : FOR Y=0 TO 4              IA
```

```
·1610 FOR Z=0 TO 5                              JA
·1620 A(X,Y,Z,0)=X+Y+Z                          HH
·1630 IF X<>0 THEN FOR P=1 TO X : A(X,Y,Z
,P)=1 : NEXT P                                  MH
·1640 IF Y<>0 THEN FOR P=4 TO Y+3:A(X,Y,Z
,P)=1:NEXT P                                    HK
·1650 IF Z<>0 THEN FOR P=8 TO Z+7:A(X,Y,Z
,P)=1:NEXT P                                    DC
·1660 NEXT Z,Y,X                                PJ
·1670 RETURN                                    IM
·1680 :REM - - - - - - - - - - - - - - - -      EO
·1690 REM  FILL MOVE ARRAY FROM FILE            MC
·1700 :REM - - - - - - - - - - - - - - - -      EO
·1710 INPUT"LOAD FROM WHAT FILE";F$             GH
·1720 OPEN 1,8,8,F$+",S,R" : GOSUB 2500         IH
·1730 FOR I=0 TO 3 : FOR J=0 TO 4               DI
·1740 FOR K=0 TO 5 : FOR P=0 TO 12              ID
·1750 INPUT#1,A(I,J,K,P)                        CD
·1760 NEXT P,K,J,I : CLOSE 1                    EM
·1770 RETURN                                    IM
·1780 :REM - - - - - - - - -                    CL
·1790 REM    USER PLAYS                         GF
·1800 :REM - - - - - - - - -                    CL
·1810 B(1)=X : B(2)=Y : B(3)=Z                  OP
·1820 FOR J=1 TO 5 : PRINT SP$ :NEXT J          DN
·1830 LL=10 : GOSUB 2400                        DB
·1840 PRINT "ROW 1 =";X                         BL
·1850 PRINT "ROW 2 =";Y                         DH
·1860 PRINT "ROW 3 =";Z                         DL
·1870 PRINT "[3" "]<PRESS M TO GO TO MENU
>"                                              DG
·1880 PRINT "[DOWN]"SP$"[UP]"                   KB
·1890 INPUT "PICK FROM WHICH ROW";RW$           MG
·1900 :REM  M = BACK TO MENU & RESTART          JP
·1910 IF RW$="M" THEN 160                       MC
·1920 RW=VAL(RW$)                               EE
·1930 IF RW<1 OR RW>3 THEN 1820                 NI
·1940 :REM  0 = EMPTY ROW                       LO
·1950 IF B(RW)=0 THEN 1820                      BH
·1960 LL=17 : GOSUB 2400                        FA
·1970 PRINT SP$"[UP]"                           DE
·1980 PRINT "TAKE HOW MANY FROM ROW";RW;        CK
·1990 INPUT TK$ : IF TK$="M" THEN 160           IF
·2000 TK=VAL(TK$)                               EA
·2010 :REM  TAKE TOO MANY?                      HK
·2020 IF TK>B(RW) THEN 1960                     JM
·2030 :REM  CHOOSE A DIFFERENT ROW              EA
·2040 IF TK<1 THEN 1820                         IO
·2050 B(RW)=B(RW)-TK                            GG
·2060 X=B(1) : Y=B(2) : Z=B(3)                  NJ
·2070 LL=5 : GOSUB 2400                         BG
·2080 PRINT "USER:  ";X;Y;Z                     AN
·2090 RETURN                                    IM
·2100 :REM - - - - - - - - - - -                HC
·2110 REM   READ KEYBOARD                       AI
·2120 :REM - - - - - - - - - - -                HC
·2130 PRINT : PRINT SP$,SP$ : PRINT L$          HP
·2140 PRINT "PRESS <C> FOR RANDOM COMPUTE
R TO PLAY[15" "]AGAINST ELMAC"                  KD
·2150 PRINT"PRESS <P> TO PLAY AGAINST ELM
AC[27" "]"                                      DI
·2160 PRINT "PRESS <L> TO LOAD THE MOVE A
RRAY FROM[15" "]DISK"                           JK
·2170 PRINT "PRESS <D> TO DISPLAY THE MOV
E ARRAY[22" "]"                                 KB
·2180 PRINT "PRESS <S> TO SAVE THE MOVE A
RRAY TO DISK" : PRINT : PRINT                   ED
·2190 GET R$ : IF R$="" THEN 2190               LK
·2200 PRINT R$                                  BG
·2205 :REM  MODE CHANGE                         IL
·2210 MDCHG=TRUE                                EJ
·2220 :REM  RANDOM COMPUTER PLAYS               HM
·2230 IF R$="C" THEN PL=1 : RETURN              CK
·2240 :REM  USER PLAYS                          KA
·2250 IF R$="P" THEN PL=2 : RETURN              CO
·2255 :REM  DISPLAY THE MOVE ARRAY              IM
·2260 IF R$<>"D" THEN 2290                      GJ
·2270 IF F0=0 THEN PRINT "THINKING[3"."]"
 : GOSUB 1580 : F0=1                            OK
·2280 GOSUB 1360 : RETURN                       DK
·2285 :REM  SAVE THE MOVE ARRAY                 JJ
·2290 IF R$="S" THEN GOSUB 1450 :RETURN         BP
·2300 :REM  LOAD THE MOVE ARRAY                 FK
·2310 IF R$="L" THEN GOSUB 1690 : F0=1 :
RETURN                                          KD
·2320 GOTO 2130 :REM  BAD KEY                   LI
·2400 REM  LOCATE CURSOR AT COLUMN ONE OF
  LINE LL (LL=1 TO 24)                          MJ
·2410 POKE 214,LL-1 : PRINT                     NE
·2420 RETURN                                    IM
·2500 REM  CHECK DISK ERROR STATUS              OE
·2510 INPUT#15,EN,EM$,ET,ES                     BD
·2520 IF EN<20 THEN 2550                        KP
·2530 PRINT EN;EM$;ET;ES : CLOSE 15 : CLO
SE 1 : GOTO 120                                 BH
·2540 :REM  NO ERROR                            OM
·2550 RETURN                                    IM
```

TREASURE WHEEL
FROM PAGE 31

```
·10 REM ********************************        CE
·20 REM **      TREASURE  WHEEL        **       PF
·30 REM ***           BY              ***       CO
·40 REM ****    BOB BLACKMER         ****       DG
·50 REM ***     6118 S. BRIAR BAYOU   ***       NF
·60 REM **      HOUSTON, TX   77072    **       EP
·70 REM ********************************        CE
·80 DATA100,1000,-1,400,200,500,0,300,100
,1000,TILT,400,200,500,LOSE A TURN,300          BP
·90 DATA PLAYER #,ENTER LETTER,F1 TO SPIN
,F3 TO SOLVE,!!WAY TO GO!!                      PJ
·100 DATA PERSON,PLACE,THING,PHRASE,TITLE CJ
·110 REM **** DATA FOR PERSON ****              HB
·120 DATA ALEXANDER THE GREAT,CHRISTOPHER
```

```
 COLUMBUS,ASTRONAUT,POSTMASTER,MAGELLAN  MN
·130 DATA RONALD REAGAN,PETE ROSE,SYLVEST
ER STALLONE,JIMMY CARTER,JOHNNY CARSON  JG
·140 DATA KERMIT THE FROG,ELVIS PRESLEY,N
UCLEAR SCIENTIST,REGISTERED NURSE  PJ
·150 DATA DOLLY PARTON,CAPTAIN KANGAROO,J
OHN GLENN,ROGER STAUBACH,JOHN BYNER  PJ
·160 DATA BURT REYNOLDS  CM
·170 REM **** DATA FOR PLACE  ****  EL
·180 DATA WEST GERMANY,UNITED STATES OF A
MERICA,ASTRODOME,NEW ENGLAND,KENTUCKY  AJ
·190 DATA HAWAIIAN ISLANDS,GREAT SMOKY MO
UNTAINS,NORTH DAKOTA,CANDY STORE  CB
·200 DATA MOVIE THEATER,WEST POINT,BEAUTY
 PARLOR,SHOPPING MALL,BOWLING ALLEY  BG
·210 DATA MASSACHUSETTS,HOUSTON,DISNEYLAN
D,NIAGARA FALLS,MOUNT RUSHMORE  PA
·220 DATA CENTRAL AMERICA  GH
·230 REM **** DATA FOR THING  ****  KC
·240 DATA LOOSELEAF PAPER,ASHTRAY,TELEVIS
ION SET,COFFEE TABLE,FOUNTAIN PEN  GC
·250 DATA THREE PIECE SUIT,LAWNMOWER,COMP
UTER,MAGAZINE,STEERING WHEEL  DD
·260 DATA BARBECUE GRILL,AUTOMOBILE,MOTOR
CYCLE,PARKING GARAGE,WATER COOLER  LG
·270 DATA HARDCOVER BOOK,COFFEE MUG,BEER
STEIN,STEREO RADIO,ROLLTOP DESK  MG
·280 REM **** DATA FOR PHRASE ****  LJ
·290 DATA NEVER ON SUNDAY,ONCE IN A BLUE
MOON,THE EARLY BIRD GETS THE WORM  BC
·300 DATA NEW AND IMPROVED,SHARP AS A TAC
K  GL
·310 DATA A STITCH IN TIME SAVES NINE,SWE
ET LAND OF LIBERTY,PAR FOR THE COURSE  GO
·320 DATA PRETTY AS A PICTURE,NUTTY AS A
FRUITCAKE,OUT ON A LIMB  DB
·330 DATA TOO CLOSE FOR COMFORT,FOLLOW TH
E YELLOW BRICK ROAD,WATCH YOUR STEP  BJ
·340 DATA DAY IN AND DAY OUT,ONCE UPON A
TIME,TO EACH HIS OWN  GK
·350 DATA A WATCHED POT NEVER BOILS,APRIL
 SHOWERS BRING MAY FLOWERS  BO
·360 DATA UP A CREEK WITHOUT A PADDLE  CB
·370 REM **** DATA FOR TITLE  ****  GO
·380 DATA THE WALL STREET JOURNAL,SPEAKER
 OF THE HOUSE,GHOSTBUSTERS,HEAD FOREMAN  IB
·390 DATA MASTER OF CEREMONIES,SUPREME CO
URT JUSTICE,NEW YORK TIMES,RAMBO  AF
·400 DATA QUEEN OF ENGLAND,CANNONBALL RUN
,INTERNAL REVENUE SERVICE,STAR WARS  DN
·410 DATA GONE WITH THE WIND,MAJOR LEAGUE
 BASEBALL,VICE PRESIDENT,FAMILY CIRCLE  ED
·420 DATA UNITED STATES SENATOR,GOVERNOR
OF MINNESOTA,SPORTS ILLUSTRATED  PF
·430 DATA DUKE OF EARL  HE
·440 SC(3)=0:SC(4)=0:SC(5)=0:SC(6)=0:A=1  JP
·450 DIMWH(8),WH$(13),U$(26),CG$(5):FORL=
1TO126:READC$:NEXT:GOSUB490:C$=""  DG
·460 RESTORE:FORL=1TO8:READWH(L):NEXT  DN
·470 FORL=1TO13:READWH$(L):NEXT:FORL=1TO5
:READCG$(L):NEXT  IA
·480 GOTO540  CN
·490 CK=0:PRINT"[CLEAR]LOADING ML DATA[3"
."]":FORL=828TO980:READA:POKEL,A:CK=CK+A
:NEXT  HN
·500 IFCK<>20211THENPRINT"[DOWN]ERROR- CH
ECK ML DATA STATEMENTS":END  DE
·510 CK=0:PRINT"[CLEAR]LOADING SPRITE DAT
A":FORL=12288TO12864:READA:POKEL,A:CK=CK
+A:NEXT  ON
·520 IFCK<>9993THENPRINT"[DOWN]ERROR- CHE
CK SPRITE DATA STATEMENTS":END  PG
·530 A=1:RETURN  PA
·540 V=53248:S=54272:P=192:Q=193:PR=65520
:N=781:E=RND(-TI):SP$="[13" "]"  GF
·550 GOSUB1420:X=9:Y=24:U=1:SC(1)=0:SC(2)
=0:FORL=1TO26:U$(L)="":NEXT  JN
·560 POKEV+39,0:POKEV+40,7:POKEV+41,7:POK
EV+3,108:POKEV+5,108:POKEV+2,114  NB
·570 POKEV+4,139:POKEV,116:POKEV+1,101:PO
KEV+21,7  GN
·580 POKE2041,P:POKE2042,Q:POKE2040,Q+7  FF
·590 FORL=STOS+24:POKEL,0:NEXT:POKES+24,1
43:POKES+18,129:POKES+14,129  KE
·600 CL=FRE(0):REM **** SPIN ??? ****  IP
·610 POKEN+2,0:POKEN,14:POKEN+1,26:SYSPR:
PRINT"[CYAN]"WH$(9);A  AC
·620 POKEN+2,0:POKEN,16:POKEN+1,26:SYSPR:
PRINT"[CYAN]"WH$(11)  CK
·630 POKEN+2,0:POKEN,18:POKEN+1,26:SYSPR:
PRINT"[CYAN]"WH$(12)  HN
·640 POKE198,0:WAIT198,1:GETAN$:IFAN$<>"[
F1]"ANDAN$<>"[F3]"THEN640  JC
·650 IFAN$=CHR$(134)THEN850  DP
·660 GOSUB1080:GOSUB910:IFB=3THENGOSUB110
0:A=A+1:IFA=3THENA=1  BD
·670 IFB=3THENGOSUB830:GOTO600  OO
·680 IFB=7THENFORTT=1TO3:GOSUB810:NEXT:A=
A+1:IFA=3THENA=1  FL
·690 IFB=7THEN600  PO
·700 REM **** ENTER YOUR LETTER ****  JD
·710 POKEN+2,0:POKEN,12:POKEN+1,29:SYSPR:
PRINTWH$(B):GOTO720  AL
·720 POKEN+2,0:POKEN,14:POKEN+1,26:SYSPR:
PRINTWH$(9);A  NJ
·730 POKEN+2,0:POKEN,16:POKEN+1,25:SYSPR:
PRINTWH$(10)  EG
·740 POKEN+2,0:POKEN,18:POKEN+1,28:SYSPR:
INPUTA$:IFASC(A$)<65ORASC(A$)>91THEN740  DC
·750 IFLEN(A$)>1THEN740  DF
·760 GOSUB1080:L$=A$:GOSUB1000:IFW=1THENW
=0:GOTO710  LO
·770 GOSUB1180:IFG=0THENGOSUB810:GOTO790  IC
·780 IFG>0THENGOSUB1100:GOSUB1230:A=A-1  IO
·790 A=A+1:IFA=3THENA=1  MD
·800 GOTO600  CC
```

```
·810 POKES,180:POKES+1,5:POKES+5,14:POKES
+6,146:POKES+4,17:FORZZ=1TO300:NEXT       BP
·820 POKES+4,16:RETURN                    CD
·830 POKES,180:POKES+1,5:POKES+5,14:POKES
+6,146:POKES+4,17:FORZZ=255TO1STEP-1      GI
·840 POKES+1,ZZ:NEXT:POKES+4,16:RETURN    BF
·850 REM **** SOLVE PUZZLE ****           IB
·860 POKEN+2,0:POKEN,4:POKEN+1,6:SYSPR:IN
PUTA$                                     MO
·870 IFA$=B$THENSC(A)=SC(A)+100:GOSUB1130
:GOTO1270                                 HB
·880 IFA$<>B$THENA$="":POKEN+2,0:POKEN,4:
POKEN+1,6:SYSPR:PRINTSP$SP$SP$            MD
·890 GOSUB810:GOSUB810:A=A+1:IFA=3THENA=1 LA
·900 GOTO600                              CC
·910 REM **** SPIN THE WHEEL ****         CH
·920 POKES+1,10:POKES+5,5:POKES+6,67      PF
·930 FORLO=1TOINT(RND(1)*7)+4:POKES+4,33  ND
·940 FORL=114TO89STEP-6:POKEV+2,L:POKEV+4
,L+25:NEXT:P=P+2:IFP=200THENP=192:POKES+
4,32                                      MC
·950 POKE2041,P:POKES+4,33                HL
·960 FORL=114TO89STEP-6:POKEV+4,L:POKEV+2
,L+25:NEXT:Q=Q+2:IFQ=201THENQ=193:POKES+
4,32                                      BK
·970 POKE2042,Q:NEXTLO:E=INT(RND(1)*7)+3:
FORL=1TOE:GOSUB990:SYS828:NEXT            CI
·980 B=PEEK(684)-191:RETURN               NP
·990 POKES+1,10:POKES+5,5:POKES+6,67:POKE
S+4,33:POKES+4,32:RETURN                  DD
·1000 REM **** USED LETTERS ****          FC
·1010 T=1                                 DA
·1020 IFU$(T)=L$THENW=1:RETURN            FL
·1030 IFU$(T)<>L$THENT=T+1:IFT=27THEN1050 GF
·1040 GOTO1020                            FA
·1050 U$(U)=L$:U=U+1:POKEN+2,0:POKEN,X:PO
KEN+1,Y:SYSPR:PRINTL$:Y=Y+1:IFY=37THENX=
10:Y=24                                   NN
·1060 IFX=10ANDY=37THENEND                DE
·1070 RETURN                              IM
·1080 REM **** CLEAR PLAYER AREA ****     IJ
·1090 FORL=12TO18STEP2:POKEN+2,0:POKEN,L:
POKEN+1,24:SYSPR:PRINTSP$:NEXT:RETURN     FB
·1100 REM **** PLAYERS SCORE ****         KF
·1110 FORL=1TOG:SC(A)=SC(A)+WH(PEEK(684)-
191):NEXT                                 JF
·1120 IFB=3THENSC(A)=0                    GE
·1130 IFA=1THENPOKEN+2,0:POKEN,21:POKEN+1
,32:SYSPR:PRINT"[6" "]"                   LC
·1140 IFA=1THENPOKEN+2,0:POKEN,21:POKEN+1
,32:SYSPR:PRINTSC(A)                      AM
·1150 IFA=2THENPOKEN+2,0:POKEN,23:POKEN+1
,32:SYSPR:PRINT"[6" "]"                   GJ
·1160 IFA=2THENPOKEN+2,0:POKEN,23:POKEN+1
,32:SYSPR:PRINTSC(A)                      HB
·1170 RETURN                              IM
·1180 YY=8:G=0:PRINT"[CYAN]":FORL=1TOLEN(
B$):K$=MID$(B$,L,1)                       IM
·1190 IFL$=K$THENPOKEN+2,0:POKEN,2:POKEN+
1,YY:SYSPR:PRINTL$:GOSUB1220:YY=YY+1:G=G
+1:NEXT                                   KF
·1200 IFL$<>K$THENYY=YY+1:NEXT            EC
·1210 RETURN                              IM
·1220 POKES+1,129:POKES+5,9:POKES+6,5:POK
ES+4,17:FORZZ=1TO300:NEXT:POKES+4,16:RET
URN                                       KN
·1230 K=1112:L=1:REM * LETTERS FILLED? *  ML
·1240 IFPEEK(K)=32ANDMID$(B$,L,1)<>" "THE
NRETURN                                   ON
·1250 K=K+1:L=L+1:IFL>LEN(B$)THENSC(A)=SC
(A)+100:GOTO1270                          GL
·1260 GOTO1240                            FA
·1270 PRINT"[CLEAR][HOME][DOWN]"TAB(11)"P
UZZLE IS SOLVED":PRINTTAB(12)WH$(13):PRI
NT"[3"[DOWN]"]"TAB(11)WH$(9)A"WINS!"      HJ
·1280 IFA=1THENSC(3)=SC(3)+1:SC(5)=SC(5)+
SC(1)                                     GK
·1290 IFA=2THENSC(4)=SC(4)+1:SC(6)=SC(6)+
SC(2)                                     JC
·1300 POKE53269,0                         PL
·1310 PRINT"[DOWN]"TAB(8)"GAMES [RVSON]WO
N[RVSOFF]"TAB(19)"TOTAL [RVSON]POINTS[RV
SOFF]"                                    HE
·1320 PRINT"[DOWN]"TAB(4);WH$(9)"1:"SC(3)
TAB(25)SC(5)                              NE
·1330 PRINT"[DOWN]"TAB(4);WH$(9)"2:"SC(4)
TAB(25)SC(6)                              FN
·1340 IFSC(3)+SC(4)=10THENPRINT"[DOWN][DO
WN]"TAB(11)"ANOTHER SET (Y/N)?":CP=1:GOT
O1360                                     BE
·1350 PRINT"[DOWN][DOWN]"TAB(11)"PLAY AGA
IN (Y/N)?"                                OK
·1360 FORL=10TO150STEP10:POKES,180:POKES+
1,L:POKES+5,14:POKES+6,146:POKES+4,17     KM
·1370 FORZZ=200TO225:POKES+1,ZZ:NEXT:NEXT
:POKES+4,16                               DJ
·1380 POKE198,0:WAIT198,1:GETB$:IFB$<>"Y"
ANDB$<>"N"THEN1380                        BE
·1390 IFCP=1ANDB$="Y"THENPOKE53269,0:RUN  CP
·1400 IFB$="Y"THENPOKE53269,0:RESTORE:GOT
O460                                      EF
·1410 END                                 IC
·1420 PRINT"[CLEAR][HOME][5"[DOWN]"]"     NI
·1430 PRINTTAB(7)"[BLACK][12"[c P]"]"     ID
·1440 PRINTTAB(6)"[YELLOW][RVSON][sEP][5"
 "][RVSOFF]  [RVSON][5" "][c *][RVSOFF][
4" "][CYAN]USED  LETTERS"                 OP
·1450 PRINTTAB(5)"[YELLOW][RVSON][sEP][6"
 "][RVSOFF]  [YELLOW][RVSON][6" "][c *][
RVSOFF][3" "][BLACK][13"[s C]"]"          LC
·1460 PRINTTAB(4)"[YELLOW][RVSON][sEP][7"
 "][RVSOFF][BLACK]  [YELLOW][RVSON][7" "
][c *]"                                   AB
·1470 PRINTTAB(3)"[YELLOW][RVSON][sEP][4"
 "][RVSOFF][sEP][GREEN][RVSON][8" "][YEL
LOW][RVSOFF][c *][RVSON][4" "][c *]"      GM
```

To enter ARENA... it is necessary to use our *Flankspeed* machine language entry program. Carefully read the introduction to *Flankspeed* on page 98 of this magazine.

```
·1480 PRINTTAB(3)"[YELLOW][RVSON][4" "][R
VSOFF][sEP][GREEN][RVSON][6" "][GREEN][R
VSON][4" "][YELLOW][RVSOFF][c *][RVSON][
4" "][RVSOFF][BLACK] [13"[s C]"]"        AL
·1490 PRINTTAB(3)"[YELLOW][RVSON][4" "][G
REEN][8" "][RVSON][4" "][YELLOW][4" "]"  NI
·1500 PRINTTAB(3)"[YELLOW][RVSON][4" "][G
REEN][5" "][BLACK][RVSOFF][s U][s I][GRE
EN][RVSON][5" "][YELLOW][4" "]"          FE
·1510 PRINTTAB(3)"[YELLOW][RVSON][4" "][G
REEN][5" "][BLACK][RVSOFF][s J][s K][GRE
EN][RVSON][5" "][YELLOW][4" "]"          JH
·1520 PRINTTAB(3)"[YELLOW][RVSON][4" "][G
REEN][8" "][RVSON][4" "][YELLOW][4" "]"  NI
·1530 PRINTTAB(3)"[YELLOW][RVSON][4" "][c
 *][GREEN][6" "][GREEN][RVSON][4" "][YEL
LOW][sEP][4" "]"                         FD
·1540 PRINTTAB(3)"[YELLOW][c *][RVSON][4"
 "][c *][GREEN][8" "][YELLOW][sEP][4" "]
[RVSOFF][sEP]"                           OJ
·1550 PRINTTAB(4)"[YELLOW][c *][RVSON][16
" "][RVSOFF][sEP]"                       JH
·1560 PRINTTAB(5)"[YELLOW][c *][RVSON][3"
 "]CATEGORY[3" "][RVSOFF][sEP]"          KH
·1570 PRINTTAB(6)"[YELLOW][c *][RVSON][5"
 "][s M][s N][5" "][RVSOFF][sEP]"        GI
·1580 PRINTTAB(7)"[BLACK][c A][10"[s C]"]
[c S][3" "][CYAN]"WH$(9)"1:":PRINTTAB(7)
"[BLACK][s B][10" "][s B]"               LM
·1590 PRINTTAB(7)"[BLACK][c Z][10"[s C]"]
[c X][3" "][CYAN]"WH$(9)"2:"             CJ
·1600 Z=INT(RND(1)*5)+1:POKEN+2,0:POKEN,2
2:POKEN+1,10:SYSPR:PRINT"[RVSON][CYAN]"C
G$(Z)                                    GA
·1610 K=Z-1:IFK=0THENFORL=1TOINT(RND(1)*2
0)+1:READB$:NEXT:YY=8:GOTO1630           BI
·1620 FORL=1TOK*20:READB$:NEXT:FORL=1TOIN
T(RND(1)*20)+1:READB$:NEXT:YY=8          CO
·1630 K=LEN(B$):FORL=1TOK                GK
·1640 IFMID$(B$,L,1)<>" "THENPOKEN+2,0:PO
KEN,3:POKEN+1,YY:SYSPR:PRINT"[BLACK]-":Y
Y=YY+1:NEXT:GOTO1660                     HL
·1650 POKEN+2,0:POKEN,3:POKEN+1,YY:PRINT"
 ":YY=YY+1:NEXT                          KO
·1660 RETURN                             IM
·1670 REM **** ML DATA ****              IJ
·1680 DATA 162,115,160,140,142,2,208,140,
4,208,136,202,224,90,208,9               KF
·1690 DATA 202,202,202,142,2,208,76,91,3,
32,182,3,76,64,3,238                     JH
·1700 DATA 249,7,238,249,7,173,249,7,201,
200,208,5,169,192,141,249                DP
·1710 DATA 7,173,27,212,201,128,16,56,162
,115,160,140,142,4,208,140               GJ
·1720 DATA 2,208,136,202,224,90,208,3,76,
141,3,32,182,3,76,120                    OD
·1730 DATA 3,202,202,202,142,4,208,238,25
0,7,238,250,7,173,250,7                  OC
·1740 DATA 201,201,208,5,169,193,141,250,
7,173,249,7,141,172,2,96                 OE
·1750 DATA 173,250,7,141,172,2,96,234,234
,234,169,3,141,170,2,169                 AM
·1760 DATA 255,141,168,2,206,168,2,173,16
8,2,201,0,208,246,206,170                GD
·1770 DATA 2,173,170,2,201,0,208,231,96  DK
·1780 REM **** SPRITE DATA ****          LF
·1790 DATA 0,16,0,48,16,12,0,80,0,0,16,0,
0,16,0,0,16,0,0,84,0,0,0,0               HH
·1800 DATA 0,84,0,1,1,0,1,1,0,1,1,0,1,1,0
,0,84,0,0,0,0,0,84,0                     OC
·1810 DATA 1,1,0,1,1,0,1,1,0,1,1,0,0,84,0
,0,0,0,0,48,16,12,0,80                   KP
·1820 DATA 0,0,16,0,0,16,0,0,84,0,0,0,0,0
,84,0,1,1,0,1,1,0,0,84                   KL
·1830 DATA 0,0,0,0,0,84,0,1,1,0,1,1,0,0,8
4,0,0,0,0,0,84,0,1,1                     EC
·1840 DATA 0,1,1,0,0,84,0,0,0,0,0,48,84,1
2,0,16,0,0,16,0,0,16,0,0                 BA
·1850 DATA 0,0,0,40,0,0,16,0,0,16,0,0,40,
0,0,0,0,0,64,0,0,64,0,0                  DF
·1860 DATA 64,0,0,84,0,0,0,0,0,84,0,0,16,
0,0,16,0,0,16,0,0,0,0,0                  CA
·1870 DATA 0,4,0,48,20,12,0,36,0,0,132,0,
1,4,0,1,85,0,0,4,0,0,0,0                 IH
·1880 DATA 0,84,0,1,1,0,1,1,0,1,1,0,1,1,0
,0,84,0,0,0,0,0,84,0                     OC
·1890 DATA 1,1,0,1,1,0,1,1,0,1,1,0,0,84,0
,0,0,40,0,48,130,12,0,2                  DK
·1900 DATA 0,0,4,0,0,40,0,0,128,0,1,85,0,
0,0,0,0,84,0,1,1,0,1,1                   MA
·1910 DATA 0,1,1,0,1,1,0,0,84,0,0,0,0,0,8
4,0,1,1,0,1,1,0,1,1                      CD
·1920 DATA 0,1,1,0,0,84,0,0,0,170,0,48,12
8,12,0,168,0,0,2,0,0,2,0,0               NG
·1930 DATA 130,0,0,84,0,0,0,0,0,84,0,1,1,
0,1,1,0,1,1,0,1,1,0,0                    FO
·1940 DATA 84,0,0,0,0,0,84,0,1,1,0,1,1,0,
1,1,0,1,1,0,0,84,0,0                     CC
·1950 DATA 133,74,149,132,72,16,132,74,14
8,132,64,144,132,64,144,165              FJ
·1960 DATA 74,149,0,0,0,0,40,0,0,68,0,0,1
70,0,0,130,0,0,130,0,0,130,0,0           BE
·1970 DATA 0,0,170,74,145,34,72,153,34,73
,21,34,74,21,34,73                       MB
·1980 DATA 19,33,136,145,0,0,0,0,0,170,0,
49,1,12,0,1,0,0,21,0,0,1,0,1             JJ
·1990 DATA 1,0,0,170,0,0,0,0,0,84,0,1,1,0
,1,1,0,1,1,0,1,1,0,0                     LJ
·2000 DATA 84,0,0,0,0,0,84,0,1,1,0,1,1,0,
1,1,0,1,1,0,0,84,0,0                     CC
·2010 DATA 0,126,0,0,60,0,0,60,0,0,24,0,0
,24,0,0,24,0,0,24,0,0,24,0               GE
```

```
·2020 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0                         FG
·2030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,44                                        MK
```

ARENA
FROM PAGE 92

Starting address in hex: C000
Ending address in hex: CACF
SYS to Start: 49152

```
C000: A0 00 B9 D0 C6 99 00 08 93
C008: B9 D0 C7 99 00 09 B9 D0 87
C010: C8 99 00 0A B9 D0 C9 99 6A
C018: 00 0B C8 D0 E5 A9 00 8D D9
C020: 21 D0 8D 20 D0 A9 9A 20 F4
C028: D2 FF A9 93 20 D2 FF A0 CB
C030: 0F A2 0B 18 20 F0 FF A0 B6
C038: 00 B9 95 C6 20 D2 FF C8 0A
C040: C9 44 D0 F5 A0 0A A2 0D 6F
C048: 18 20 F0 FF A0 00 B9 9F 6B
C050: C6 20 D2 FF C8 C9 3F D0 AC
C058: F5 A9 00 85 C6 A5 C6 F0 A1
C060: FC AD 77 02 C9 53 F0 0B 9D
C068: C9 4D F0 0E C9 46 F0 11 90
C070: 4C 59 C0 A9 0C 85 17 4C 75
C078: 85 C0 A9 0A 85 17 4C 85 E0
C080: C0 A9 08 85 17 A9 00 8D C6
C088: 18 D4 A9 7F 8D 13 D4 A9 BD
C090: 7F 8D 14 D4 A9 02 8D 0F CE
C098: D4 A9 81 8D 12 D4 20 7A A7
C0A0: C5 20 40 C5 20 6B C5 20 FD
C0A8: B9 C5 20 DC C4 A0 00 B9 44
C0B0: C9 C6 99 28 D0 C8 C0 07 64
C0B8: D0 F5 A9 02 85 14 78 AD EA
C0C0: 37 C6 8D 14 03 AD 38 C6 10
C0C8: 8D 15 03 58 20 F0 C0 20 B8
C0D0: 54 C2 20 20 C3 20 54 C2 23
C0D8: 20 20 C3 20 9F C2 20 20 9F
C0E0: C3 20 BA C1 20 20 C3 20 65
C0E8: FB C0 20 FB C2 4C CC C0 5E
C0F0: A6 17 A4 15 88 D0 FD CA 8A
C0F8: D0 F8 60 AD 01 DC 29 1F F6
C100: 49 1F C9 01 D0 07 85 0B 9B
C108: A9 00 85 03 60 C9 02 D0 37
C110: 07 85 0B A9 00 85 03 60 3A
C118: C9 04 D0 07 85 03 A9 00 EF
C120: 85 0B 60 C9 08 D0 07 85 40
C128: 03 A9 00 85 0B 60 C9 10 9F
C130: 90 03 20 DE C1 60 BD 00 A2
C138: D0 C9 1A B0 15 C0 00 F0 64
C140: 0C A9 AF 99 03 00 AD 1B 0B
C148: D4 99 0B 00 60 A9 08 85 59
C150: 03 60 DE 00 D0 60 BD 00 81
C158: D0 C9 FF 90 15 C0 00 F0 4A
C160: 0C A9 55 99 03 00 AD 1B D0
C168: D4 99 0B 00 60 A9 04 85 75
C170: 03 60 FE 00 D0 60 BD 01 C2
C178: D0 C9 33 B0 15 C0 00 F0 BD
C180: 0C A9 AF 99 0B 00 AD 1B 53
C188: D4 99 03 00 60 A9 02 85 8B
C190: 0B 60 DE 01 D0 60 BD 01 CB
C198: D0 C0 00 F0 10 C9 EE 90 74
C1A0: 15 A9 55 99 0B 00 AD 1B 22
C1A8: D4 99 03 00 60 C9 F2 90 C7
C1B0: 05 A9 01 85 0B 60 FE 01 51
C1B8: D0 60 A0 00 A2 00 A5 03 D5
C1C0: F0 08 C9 04 F0 14 20 56 03
C1C8: C1 60 A5 0B F0 07 C9 01 5E
C1D0: F0 04 20 96 C1 60 20 76 35
C1D8: C1 60 20 36 C1 60 85 FB F4
C1E0: AD 15 D0 29 02 D0 6C AD 8A
C1E8: 00 D0 8D 02 D0 AD 01 D0 99
C1F0: 8D 03 D0 A5 FB C9 11 F0 BF
C1F8: 1E C9 12 F0 2C C9 14 F0 DE
C200: 3A C9 18 D0 4E AD 02 D0 BB
C208: C9 EE B0 47 69 10 8D 02 C1
C210: D0 A9 08 85 13 D0 34 AD DD
C218: 03 D0 C9 35 90 35 E9 02 9C
C220: 8D 03 D0 A9 01 85 13 D0 95
C228: 22 AD 03 D0 C9 E6 B0 23 50
C230: 69 07 8D 03 D0 A9 02 85 33
C238: 13 D0 10 AD 02 D0 C9 1D 93
C240: 90 11 E9 03 8D 02 D0 A9 D8
C248: 04 85 13 AD 15 D0 09 02 83
C250: 8D 15 D0 60 AD 15 D0 29 E0
C258: 02 D0 01 60 A5 13 C9 02 11
C260: F0 13 C9 04 F0 1A C9 08 0F
C268: F0 21 AD 03 D0 C9 32 F0 E8
C270: 25 CE 03 D0 60 AD 03 D0 1A
C278: C9 EE F0 1A EE 03 D0 60 5F
C280: AD 02 D0 C9 19 F0 0F CE B2
C288: 02 D0 60 AD 02 D0 C9 FF 06
C290: F0 04 EE 02 D0 60 AD 15 6A
C298: D0 29 FD 8D 15 D0 60 AD 12
C2A0: 15 D0 29 FC F0 3C A0 02 7C
C2A8: AD 15 D0 39 41 C6 F0 2C 9A
C2B0: B9 49 C6 AA B9 03 00 C9 AB
C2B8: AA B0 0A C9 55 90 09 20 F6
C2C0: 36 C1 4C C8 C2 20 56 C1 C8
C2C8: B9 0B 00 C9 AA B0 0A C9 86
C2D0: 55 90 09 20 76 C1 4C DC 41
C2D8: C2 20 96 C1 C8 C0 08 D0 76
C2E0: C7 60 A9 00 8D 18 D4 A5 D2
C2E8: 15 C9 9C 90 04 E9 0A 85 72
C2F0: 15 20 37 C4 20 B6 C4 20 DD
C2F8: DC C4 60 A5 02 F0 20 C6 7A
C300: 02 D0 1C A9 00 8D 18 D4 13
C308: A0 02 B9 F8 07 C9 25 D0 24
C310: 09 AD 15 D0 39 39 C6 8D 73
C318: 15 D0 C8 C0 08 D0 EB 60 AC
C320: AD 1E D0 29 01 D0 5B AD C0
C328: 15 D0 29 02 D0 01 60 A0 0C
C330: 02 AD 15 D0 39 41 C6 F0 F7
C338: 43 B9 F8 07 C9 25 F0 3C 51
```

```
C340: B9 49 C6 AA AD 02 D0 DD 13
C348: 00 D0 90 30 FD 00 D0 C9 72
C350: 12 B0 29 AD 03 D0 DD 01 9C
C358: D0 90 21 FD 01 D0 C9 0C 80
C360: B0 1A A9 25 99 F8 07 AD 41
C368: 15 D0 29 FD 8D 15 D0 A9 92
C370: 0F 8D 18 D4 A9 19 85 02 44
C378: 20 4B C4 60 C8 C0 08 D0 6B
C380: B0 60 A9 00 8D 18 D4 CE 84
C388: E9 04 A9 0F 8D 18 D4 78 22
C390: A9 31 8D 14 03 A9 EA 8D 32
C398: 15 03 58 A9 00 8D 1D D0 2E
C3A0: A9 25 8D F8 07 20 37 C4 19
C3A8: A9 00 8D 18 D4 78 AD 37 2A
C3B0: C6 8D 14 03 AD 38 C6 8D 56
C3B8: 15 03 58 AD E9 04 C9 30 BE
C3C0: F0 04 20 DC C4 60 A0 00 78
C3C8: B9 56 C6 99 EC 05 C8 C0 B4
C3D0: 09 D0 F5 A9 00 8D 15 D0 BD
C3D8: 20 8C C4 20 37 C4 A9 00 10
C3E0: 85 C6 A5 C6 D0 0A AD 01 23
C3E8: DC 29 10 F0 16 4C E2 C3 F8
C3F0: AD 77 02 C9 85 D0 05 68 A5
C3F8: 68 4C 1D C0 C9 51 D0 DE 56
C400: 4C E2 FC 20 0D C4 20 40 7E
C408: C5 20 DC C4 60 A0 00 A9 3A
C410: F0 99 EC 05 C8 C0 09 D0 EF
C418: F8 60 C6 14 D0 16 A9 02 DE
C420: 85 14 AD F8 07 C9 23 F0 45
C428: 06 EE F8 07 4C 31 EA A9 2F
C430: 20 8D F8 07 4C 31 EA A0 E6
C438: 02 98 48 A0 00 A2 00 88 E6
C440: D0 FD CA D0 FA 68 A8 88 3F
C448: D0 EF 60 A4 16 A2 03 FE C8
C450: 71 04 BD 71 04 C9 3A 90 8D
C458: 0B A9 30 9D 71 04 CA 10 2B
C460: EE 4C 68 C4 88 D0 E6 60 69
C468: A9 93 20 D2 FF A0 00 8C C5
C470: 18 D4 B9 76 C6 20 D2 FF 47
C478: C8 C9 00 D0 F5 20 37 C4 ED
C480: 20 37 C4 20 37 C4 20 37 10
C488: C4 4C E2 FC A2 FF E8 E0 E5
C490: 04 F0 22 BD 71 04 DD D9 92
C498: 05 F0 F3 90 18 AD 71 04 4E
C4A0: 8D D9 05 AD 72 04 8D DA 99
C4A8: 05 AD 73 04 8D DB 05 AD EE
C4B0: 74 04 8D DC 05 60 E6 16 F5
C4B8: A5 18 C9 2F F0 05 E6 18 64
C4C0: 4C C7 C4 A9 26 85 18 A2 A9
C4C8: 02 FE 61 05 BD 61 05 C9 1E
C4D0: 3A 90 08 A9 30 9D 61 05 81
C4D8: CA 10 EE 60 A0 02 A5 18 63
C4E0: 99 F8 07 C8 C0 08 D0 F8 D5
C4E8: A9 20 8D F8 07 A9 24 8D 9B
C4F0: F9 07 A0 02 B9 49 C6 AA 09
C4F8: B9 B9 C6 9D 00 D0 B9 C1 1D
C500: C6 9D 01 D0 C8 C0 08 D0 98
C508: EB A9 19 8D 00 D0 A9 87 46
C510: 8D 01 D0 A9 01 8D 1D D0 95
C518: AD 1E D0 A0 00 A2 00 A9 A1
C520: AF 99 03 00 AD 1B D4 99 A3
C528: 0B 00 C8 CA D0 FD C0 08 5E
C530: D0 ED A9 AF 85 03 A9 00 7A
C538: 85 0B A9 FD 8D 15 D0 60 44
C540: A9 33 8D E9 04 A9 30 8D FF
C548: 71 04 8D 61 05 8D 72 04 B5
C550: 8D 62 05 8D 73 04 8D 74 4C
C558: 04 A9 31 8D 63 05 A9 01 D7
C560: 85 16 A9 FF 85 15 A9 26 10
C568: 85 18 60 A9 30 8D D9 05 AC
C570: 8D DA 05 8D DB 05 8D DC B6
C578: 05 60 A9 93 20 D2 FF A0 AE
C580: 00 A9 0B 99 00 D8 99 00 41
C588: D9 99 00 DA 99 00 DB C8 15
C590: D0 F1 A9 01 8D 71 D8 8D 63
C598: 61 D9 8D D9 D9 8D 72 D8 ED
C5A0: 8D 62 D9 8D DA D9 8D 73 AD
C5A8: D8 8D 63 D9 8D DB D9 8D 1D
C5B0: 74 D8 8D DC D9 8D E9 D8 92
C5B8: 60 A9 00 85 FB A9 04 85 77
C5C0: FC A2 00 A0 00 A9 F0 91 2D
C5C8: FB C8 C0 1F D0 F9 18 A5 F5
C5D0: FB 69 28 85 FB A5 FC 69 EB
C5D8: 00 85 FC E8 E0 19 D0 E3 F2
C5E0: A0 21 A2 01 18 20 F0 FF 6F
C5E8: A0 00 B9 51 C6 20 D2 FF 4E
C5F0: C8 C9 45 D0 F5 A0 21 A2 F3
C5F8: 04 18 20 F0 FF A0 00 B9 80
C600: 5F C6 20 D2 FF C8 C9 53 FE
C608: D0 F5 A0 21 A2 07 18 20 72
C610: F0 FF A0 00 B9 6B C6 20 AD
C618: D2 FF C8 C9 00 D0 F5 A0 E4
C620: 21 A2 0A 18 20 F0 FF A0 B7
C628: 00 B9 71 C6 20 D2 FF C8 D5
C630: C9 00 D0 F5 60 64 C6 1A 66
C638: C4 FE FD FB F7 EF DF BF 7D
C640: 7F 01 02 04 08 10 20 40 3F
C648: 80 00 02 04 06 08 0A 0C F2
C650: 0E 53 43 4F 52 45 87 81 E4
C658: 8D 85 A0 8F 96 85 92 4C 96
C660: 49 56 45 53 53 50 52 49 D7
C668: 54 45 53 4C 45 56 45 4C CE
C670: 00 48 49 47 48 00 20 20 D1
C678: 49 20 47 49 56 45 20 55 83
C680: 50 21 21 21 20 2D 20 59 FA
C688: 4F 55 20 57 49 4E 21 21 7E
C690: 21 21 21 21 00 47 41 4D EA
C698: 45 20 53 50 45 45 44 05 75
C6A0: 53 9A 4C 4F 57 2C 05 4D FF
C6A8: 9A 45 44 49 55 4D 20 4F 28
C6B0: 52 20 05 46 9A 41 53 54 F1
C6B8: 3F 00 00 32 64 96 32 64 BB
C6C0: 96 00 00 64 64 64 AF AF E3
C6C8: AF 0A 03 04 0E 03 04 0E AC
C6D0: 00 00 00 00 00 00 00 00 D0
C6D8: 00 FF 00 00 00 00 00 00 D8
```

```
C6E0: 00 00 00 00 00 00 00 00 E0
C6E8: 00 00 00 00 00 00 00 00 E8
C6F0: 00 00 00 00 00 00 00 00 F0
C6F8: 00 00 00 00 00 00 00 00 F8
C700: 00 00 00 00 00 00 00 00 00
C708: 00 00 00 00 00 00 00 FF 08
C710: 00 00 00 60 00 00 30 00 A0
C718: 00 18 00 00 0C 00 00 06 42
C720: 00 00 00 00 00 00 00 00 20
C728: 00 00 00 00 00 00 00 00 28
C730: 00 00 00 00 00 00 00 00 30
C738: 00 00 00 00 00 00 00 00 38
C740: 00 00 00 00 00 00 00 00 40
C748: 00 00 00 00 00 00 00 00 48
C750: 18 00 00 18 00 00 18 00 98
C758: 00 18 00 00 18 00 00 18 A0
C760: 00 00 18 00 00 00 00 00 78
C768: 00 00 00 00 00 00 00 00 68
C770: 00 00 00 00 00 00 00 00 70
C778: 00 00 00 00 00 00 00 00 78
C780: 00 00 00 00 00 00 00 00 80
C788: 00 00 00 00 00 00 00 FF 88
C790: 00 00 00 06 00 00 0C 00 A2
C798: 00 18 00 00 30 00 00 60 41
C7A0: 00 00 00 00 00 00 00 00 A0
C7A8: 00 00 00 00 00 00 00 00 A8
C7B0: 00 00 00 00 00 00 00 00 B0
C7B8: 00 00 00 00 00 00 00 00 B8
C7C0: 00 00 00 00 00 00 00 00 C0
C7C8: 00 00 00 00 00 00 00 00 C8
C7D0: E0 00 00 E0 00 00 00 00 92
C7D8: 00 00 00 00 00 00 00 00 D8
C7E0: 00 00 00 00 00 00 00 00 E0
C7E8: 00 00 00 00 00 00 00 00 E8
C7F0: 00 00 00 00 00 00 00 00 F0
C7F8: 00 00 00 00 00 00 00 00 F8
C800: 00 00 00 00 00 00 00 00 00
C808: 00 00 00 00 00 00 00 FF 08
C810: CC CC 00 00 00 00 CC CC 43
C818: 00 00 00 00 CC CC 00 00 B1
C820: 00 00 CC CC 00 00 00 00 B9
C828: CC CC 00 00 00 00 CC CC 5B
C830: 00 00 00 00 00 00 00 00 30
C838: 00 00 00 00 00 00 00 00 38
C840: 00 00 00 00 00 00 00 00 40
C848: 00 00 00 00 00 00 00 00 48
C850: FF 80 00 FF 80 00 FF 80 D1
C858: 00 FF 80 00 FF 80 00 FF 59
C860: FF 00 01 FF 00 01 FF 00 62
C868: 01 FF 00 01 FF 00 01 FF 6B
C870: 00 00 00 00 00 00 00 00 70
C878: 00 00 00 00 00 00 00 00 78
C880: 00 00 00 00 00 00 00 00 80
C888: 00 00 00 00 00 00 00 BF 48
C890: 03 E0 00 03 E0 00 03 E0 3C
C898: 00 03 E0 00 FF FF 80 FF FC
C8A0: FF 80 FF FF 80 03 E0 00 85
C8A8: 03 E0 00 03 E0 00 03 E0 54
C8B0: 00 00 00 00 00 00 00 00 B0
C8B8: 00 00 00 00 00 00 00 00 B8
C8C0: 00 00 00 00 00 00 00 00 C0
C8C8: 00 00 00 00 00 00 00 00 C8
C8D0: 00 80 00 01 C0 00 03 E0 F6
C8D8: 00 07 F0 00 0F F8 00 1F F7
C8E0: FC 00 3F FE 00 7F FF 00 9B
C8E8: FF FF 80 00 00 00 00 00 69
C8F0: 00 00 00 00 00 00 00 00 F0
C8F8: 00 00 00 00 00 00 00 00 F8
C900: 00 00 00 00 00 00 00 00 00
C908: 00 00 00 00 00 00 00 FF 08
C910: FC 1F 80 7E 3F 00 3F 7E 28
C918: 00 1F FC 00 0F F8 00 07 43
C920: F0 00 0F F8 00 1F FC 00 35
C928: 3F 7E 00 7E 3F 00 FC 1F BF
C930: 80 00 00 00 00 00 00 00 B0
C938: 00 00 00 00 00 00 00 00 38
C940: 00 00 00 00 00 00 00 00 40
C948: 00 00 00 00 00 00 00 00 48
C950: 01 C0 00 0F F8 00 3F FE 58
C958: 00 3F FE 00 FF FF 80 FF 17
C960: FF 80 FF FF 80 3F FE 00 9F
C968: 3F FE 00 0F F8 00 01 C0 70
C970: 00 00 00 00 00 00 00 00 70
C978: 00 00 00 00 00 00 00 00 78
C980: 00 00 00 00 00 00 00 00 80
C988: 00 00 00 00 00 00 00 FF 88
C990: 00 80 00 03 E0 00 07 F0 EC
C998: 00 0F F8 00 1F FC 00 1F DB
C9A0: FC 00 3F FE 00 7F FF 00 5B
C9A8: 7F FF 00 FF FF 80 FF FF A8
C9B0: 80 00 00 00 00 00 00 00 31
C9B8: 00 00 00 00 00 00 00 00 B8
C9C0: 00 00 00 00 00 00 00 00 C0
C9C8: 00 00 00 00 00 00 00 00 C8
C9D0: 07 F0 00 0F F8 00 1F FC EC
C9D8: 00 3F FE 00 7F FF 00 FF 96
C9E0: FF 80 7F FF 00 3F FE 00 1F
C9E8: 1F FC 00 0F F8 00 07 F0 05
C9F0: 00 00 00 00 00 00 00 00 F0
C9F8: 00 00 00 00 00 00 00 00 F8
CA00: 00 00 00 00 00 00 00 00 00
CA08: 00 00 00 00 00 00 00 FF 08
CA10: FF FF 80 FF FF 80 FF FF 11
CA18: 80 FF FF 80 FF FF 80 FF 99
CA20: FF 80 FF FF 80 FF FF 80 A1
CA28: FF FF 80 FF FF 80 FF FF 29
CA30: 80 00 00 00 00 00 00 00 B0
CA38: 00 00 00 00 00 00 00 00 38
CA40: 00 00 00 00 00 00 00 00 40
CA48: 00 00 00 00 00 00 00 00 48
CA50: 00 80 00 01 C0 00 03 E0 76
CA58: 00 0F F8 00 3F FE 00 FF 9E
CA60: FF 80 3F FE 00 0F F8 00 27
CA68: 03 E0 00 01 C0 00 00 80 8E
CA70: 00 00 00 00 00 00 00 00 70
CA78: 00 00 00 00 00 00 00 00 78
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
CA80: 00 00 00 00 00 00 00 00 80
CA88: 00 00 00 00 00 00 00 BF 48
CA90: 00 7F 80 00 FF 80 01 FF 12
CA98: 00 03 FE 00 07 FC 00 0F AD
CAA0: F8 00 1F F0 00 3F E0 00 C9
CAA8: 7F C0 00 FF 80 00 FF 00 69
CAB0: 00 00 00 00 00 00 00 00 B0
CAB8: 00 00 00 00 00 00 00 00 B8
CAC0: 00 00 00 00 00 00 00 00 C0
CAC8: 00 00 00 00 00 00 00 00 C8
```

1541 ALIGNMENT CHECK

FROM PAGE 83

```
·10 S=53280:POKES,6:POKES+1,6                    GJ
·15 TL$="[22"[s *]"]"                            HD
·20 BL$="[25"[s *]"]"                            ED
·25 T=18:TK=T:OPEN15,8,15,"I"                    EI
·30 PRINT"[CLEAR][YELLOW][8" "][s U]"TL$"
 [s I]"                                          KJ
·35 PRINT" [7"[s *]"][s B] [WHITE]1541 AL
 IGNMENT CHECK[YELLOW] [s B][7"[s *]"]"          OD
·40 PRINT"[8" "][s J]"TL$"[s K]"                 JG
·45 PRINT"[11" "]BY  GREG GRANVILLE"             CG
·50 PRINT"[DOWN][DOWN][7" "][s U][s *][s
 *]ACTUAL HEAD POSITION[3"[s *]"][s I]"          NI
·55 PRINT"[7" "][s B][6" "][WHITE]TRACK =
 "SPC(12)"[YELLOW][s B]"                         HL
·60 PRINT"[7" "][s J]"BL$"[s K]"                 EH
·65 PRINT"[DOWN][7" "][s U][s *]POSITION
 READ FROM DISK[s *][s I]"                       LN
·70 PRINT"[7" "][s B][6" "][WHITE]TRACK =
 "SPC(12)"[YELLOW][s B]"                         HL
·75 PRINT"[7" "][s J]"BL$"[s K]"                 EH
·80 PRINT"[DOWN][7" "][s U][s *][s *]ERRO
 R STATUS[11"[s *]"][s I]"                       ID
·85 PRINT"[7" "][s B]"SPC(25)"[YELLOW][s
 B]"                                             IH
·90 PRINT"[7" "][s J]"BL$"[s K]"                 EH
·95 PRINT"[DOWN][7" "][s U][s *][s *]ALIG
 NMENT STATUS[7"[s *]"][s I]"                    JC
·100 PRINT"[7" "][s B]"SPC(25)"[YELLOW][s
 B]"                                             IH
·105 PRINT"[7" "][s J]"BL$"[s K]"                EH
·110 PRINT"[DOWN]PUSH JOYSTICK LEFT & RIG
 HT TO MOVE HEAD"                                FK
·115 PRINT" HOLD FIRE BUTTON TO READ DRIV
 E STATUS[WHITE]"                                LK
·120 :                                           DI
·125 REM **************************              OG
·130 REM *** CALCULATE T/S READ ***              AE
·135 REM **************************              OG
·140 :                                           DI
·145 PRINT#15,"M-R"CHR$(0)CHR$(28)               EC
·150 GET#15,F$:IFF$=""THENF$=CHR$(0)             BO
·155 A=ASC(F$)                                   HN
·160 BT=3ANDA                                    HB
·165 A=(159ANDA)OR(96+32*((T>17)+(T>24)+(
 T>30)))                                         LB
·170 PRINT#15,"M-W"CHR$(0)CHR$(28)CHR$(1)
 CHR$(AOR4)                                      IC
·175 :                                           DI
·180 REM ******************************          NK
·185 REM *** PRINT STATUS TO SCREEN ***          GF
·190 REM ******************************          NK
·195 :                                           DI
·200 PRINT"[HOME][7"[DOWN]"]"TAB(21)T"[LE
 FT][4" "]"                                      FM
·205 IFE<2THENPRINT"[HOME][15"[DOWN]"]"TA
 B(12)"NO ERROR DETECTED"                        DM
·210 IFE>1THENPRINT"[HOME][15"[DOWN]"]"TA
 B(12)"READ ERROR SENSED"                        AK
·215 R$=STR$(TK):IFE<1THENR$="  "                FB
·220 TR$=RIGHT$(R$,2):PRINT"[HOME][11"[DO
 WN]"]"TAB(22);TR$                               IL
·225 IFE<>0AND VAL(TR$)<(T-.5)ORVAL(TR$)>
 (T+.5)THEN245                                   NK
·230 IFE>1ANDT<>INT(T)THENPRINT"[7"[DOWN]
 "]"TAB(14)"ALIGNMENT OK ":E=0:GOTO275           KM
·235 IFE>1ANDT=INT(T)THENPRINT"[7"[DOWN]"
 ]"TAB(14)"ALIGNMENT BAD":E=0:GOTO275            MF
·240 GOTO250                                     CD
·245 PRINT"[7"[DOWN]"]"TAB(14)"ALIGNMENT
 BAD":E=0:GOTO275                                GP
·250 PRINT"[7"[DOWN]"]"TAB(14)" NOT CERTA
 IN "                                            PO
·255 REM ***************************             CE
·260 REM ****** READ JOYSTICK ******             DF
·265 REM ***************************             CE
·270 :                                           DI
·275 J=NOTPEEK(56320)AND15                       NK
·280 IFPEEK(653)THEN410                          JB
·285 IF(NOTPEEK(56320)AND16)THEN410              PA
·290 IFJ=4ANDT>1THENC=-1:GOTO340                 JE
·295 IFJ=8ANDT<35THENC=1:GOTO340                 BA
·300 IFPEEK(197)=33THENPRINT#15,"I":T=18:
 E=0:A=214:GOTO160                               EH
·305 IFPEEK(197)=23THEN520                       LM
·310 GOTO145                                     CM
·315 :                                           DI
·320 REM ****************************            GH
·325 REM *** CHANGE HEAD POSITION ***            HA
·330 REM ****************************            GH
·335 :                                           DI
·340 BT=(BT+C)AND3                               BN
·345 T=T+C*.5:IFT<1THENT=1                       PN
·350 TK=INT(T)                                   LG
·355 IFT>35THENT=35                              BD
·360 B=A AND252                                  IA
·365 C=B+BT                                      IL
·370 PRINT#15,"M-W"CHR$(0)CHR$(28)CHR$(1)
```

```
 CHR$(C)                                        MD
•375 E=0                                        CO
•380 GOTO145                                    CM
•385 :                                          DI
•390 REM ***********************                HE
•395 REM *** READ THE HEADER ***                EE
•400 REM ***********************                HE
•405 :                                          DI
•410 M$="":RESTORE                              CA
•415 FORY=1TO11:READX:M$=M$+CHR$(X):NEXT        LB
•420 PRINT#15,"M-W"CHR$(0)CHR$(3)CHR$(11)
 M$                                             KC
•425 PRINT#15,"M-W"CHR$(0)CHR$(0)CHR$(1)C
 HR$(208)                                       MH
•430 PRINT#15,"M-R"CHR$(0)CHR$(0)               PE
•435 GET#15,ER$:E=ASC(ER$+CHR$(0))              AM
•440 IFE>127THEN340                             GJ
•445 IFE<>1THEN145                              LA
•450 :                                          DI
•455 REM ****************************           GH
•460 REM *** NO ERROR/GET TRACK # ***           GE
•465 REM ****************************           GH
•470 :                                          DI
•475 PRINT#15,"M-R"CHR$(22)CHR$(0)CHR$(4)       HO
•480 FORZ=1TO2:GET#15,B$:NEXT                   DP
•485 GET#15,B$:TK=ASC(B$+CHR$(0))               OG
•490 GOTO145                                    CM
•495 :                                          DI
•500 REM ****************************           GH
•505 REM *** EXIT/DRIVE MOTOR OFF ***           DM
•510 REM ****************************           GH
•515 :                                          DI
•520 PRINT#15,"M-W"CHR$(0)CHR$(28)CHR$(1)
 CHR$(0)                                        KA
•525 FORX=1TO10:GETX$:NEXT                      OI
•530 CLOSE15:PRINT"[CLEAR]MOTOR OFF"CHR$(
 13)"PROGRAM TERMINATED":END                    JK
•535 DATA169,48,133,69,169,0,133,63,76,17
 7,243                                          OO
```

TURNKEY 64
FROM PAGE 53

Starting Address in Hex: 8000
Ending Address in Hex: 810C
Sys to start: 32768
***Flankspeed* required for entry. See page 98.**

```
8000: 09 80 5E FE C3 C2 CD 38 73
8008: 30 8E 16 D0 20 84 FF 20 72
8010: 87 FF 20 8A FF 20 81 FF E3
8018: A9 0C 8D 20 D0 A9 00 8D 83
8020: 21 D0 A9 0E 8D 86 02 A9 89
8028: 7F 8D 00 DC AD 01 DC C9 67
8030: DF D0 0E A2 20 BD CB 80 BB
8038: 9D 3C 03 CA 10 F7 4C 3C 70
8040: 03 20 53 E4 20 BF E3 A2 02
8048: 20 A9 FF 8D 04 DD 8D 05 14
8050: DD A9 19 8D 0E DD AD 0D 25
8058: DD AD 0D DD 29 01 F0 F9 E3
8060: CA D0 E6 A9 00 85 90 A9 4C
8068: 08 20 B1 FF A9 6F 20 93 0F
8070: FF A5 90 F0 07 A9 00 85 CD
8078: 90 4C 33 80 20 AE FF A9 81
8080: EA 8D 28 03 A9 08 AA A0 21
8088: 00 20 BA FF A9 04 A2 C7 7B
8090: A0 80 20 BD FF A9 00 A6 DF
8098: 2B A4 2C 20 D5 FF B0 D5 11
80A0: 20 E7 FF A5 AE A4 AF 85 D6
80A8: 2D 84 2E 20 59 A6 20 33 FB
80B0: A5 A9 00 20 90 FF 20 33 04
80B8: A5 A2 20 BD EC 80 9D 3C 26
80C0: 03 CA 10 F7 4C 3C 03 30 52
80C8: 3A 3F 2A 8D 00 DE A9 00 82
80D0: 8D 04 80 2C 04 80 30 0C CF
80D8: A0 A0 85 37 8D 83 02 84 6E
80E0: 38 8C 84 02 A9 ED 8D 28 79
80E8: 03 6C 00 A0 8D 00 DE A9 0F
80F0: 00 8D 04 80 2C 04 80 30 E3
80F8: 0C A0 A0 85 37 8D 83 02 16
8100: 84 38 8C 84 02 A9 ED 8D F4
8108: 28 03 4C AE A7 D5
```

CRABFIGHT
FROM PAGE 69

Starting Address in Hex: C000
Ending Address in Hex: CB07
***Flankspeed* required for entry. See page 98.**

```
C000: A0 00 8C 15 D0 8C 20 D0 90
C008: 8C 21 D0 B9 95 C9 20 D2 92
C010: FF C8 C0 19 90 F5 20 E4 3E
C018: FF F0 FB C9 31 90 F7 C9 52
C020: 34 B0 F3 20 D2 FF 38 E9 0E
C028: 30 8D FE 02 A0 00 B9 AE EF
C030: C9 20 D2 FF C8 C0 16 90 1D
C038: F5 20 E4 FF F0 FB C9 31 1B
C040: 90 F7 C9 33 B0 F3 20 D2 5D
C048: FF 38 E9 30 8D F0 02 C9 E4
C050: 01 D0 1D A0 00 B9 C4 C9 28
C058: 20 D2 FF C8 C0 16 90 F5 71
C060: 20 E4 FF F0 FB C9 31 90 DD
C068: F7 C9 34 B0 F3 20 D2 FF F5
C070: 20 4A C8 38 E9 30 85 FD 79
C078: A9 05 38 E5 FD 85 FD 85 4C
C080: FE A9 93 20 D2 FF A0 00 50
C088: 98 99 A7 02 C8 C0 38 90 B6
C090: F8 8D 04 D0 A9 A0 8D 08 CB
C098: 04 8D 1E 04 A2 06 8E 08 8B
C0A0: D8 8E 1E D8 A0 28 A9 A0 12
C0A8: 99 08 04 99 F8 04 99 E8 67
C0B0: 05 99 D8 06 99 1E 04 99 83
C0B8: 0E 05 99 FE 05 99 EE 06 F7
C0C0: 8A 99 08 D8 99 F8 D8 99 CA
C0C8: E8 D9 99 D8 DA 99 1E D8 69
C0D0: 99 0E D9 99 FE D9 99 EE 4D
```

```
C0D8: DA C0 F0 B0 08 98 18 69 38
C0E0: 28 A8 4C A6 C0 A0 00 A2 A8
C0E8: 06 A9 A0 99 09 04 A9 66 EF
C0F0: 99 31 04 99 59 04 99 81 D1
C0F8: 04 99 E9 05 99 51 07 99 11
C100: 79 07 99 A1 07 A9 A0 99 A6
C108: C9 07 8A 99 09 D8 A9 02 8A
C110: 99 31 D8 99 59 D8 99 81 9A
C118: D8 8A 99 E9 D9 A9 04 99 20
C120: 51 DB 99 79 DB 99 A1 DB 53
C128: 8A 99 C9 DB C8 C0 15 B0 41
C130: 03 4C E9 C0 A0 00 A9 30 A4
C138: 99 E2 05 99 32 06 A9 02 37
C140: 99 E2 D9 8D 03 DA A9 04 AF
C148: 99 32 DA 8D 53 DA C8 C0 34
C150: 03 90 E3 A0 00 B9 DA C9 C6
C158: 99 91 05 B9 DF C9 99 B2 38
C160: 05 A9 05 99 91 D9 99 B2 65
C168: D9 C8 C0 05 90 E7 A0 00 E9
C170: A2 C0 8A 99 F8 07 E8 C8 A9
C178: C0 04 90 F6 A9 C3 99 F8 C4
C180: 07 C8 C0 07 90 F8 A0 00 42
C188: B9 08 CA 99 00 30 C8 C0 68
C190: 00 D0 F5 A9 7F 8D 15 D0 F3
C198: A9 78 8D 06 D0 8D 0A D0 87
C1A0: A9 DC 8D 08 D0 8D 0C D0 F7
C1A8: A9 3B 8D 07 D0 8D 09 D0 5A
C1B0: A9 DB 8D 0B D0 8D 0D D0 0B
C1B8: A9 02 8D 27 D0 A9 04 8D 25
C1C0: 28 D0 A9 01 8D 29 D0 A9 95
C1C8: 0F 8D 2A D0 8D 18 D4 A9 84
C1D0: 05 8D 2B D0 A9 07 8D 2C C9
C1D8: D0 A9 03 8D 2D D0 A9 AA 36
C1E0: 8D 00 D0 8D 02 D0 8D 04 31
C1E8: D0 85 FC A9 53 8D 01 D0 98
C1F0: 85 FB A9 C3 8D 03 D0 A9 EA
C1F8: 8B 8D 05 D0 A9 33 8D 03 55
C200: 06 8D 53 06 AD 1E D0 20 A9
C208: 38 C8 20 4A C8 A9 15 8D 88
C210: 04 D4 A9 F0 8D 06 D4 A9 95
C218: 30 8D 01 D4 20 4A C8 EE CD
C220: CF 02 AD CF 02 C9 05 90 D0
C228: DB 20 38 C8 AD FE 02 85 59
C230: FB AD AD 02 D0 0E A5 FC 0B
C238: D0 14 AD 05 D0 C9 37 90 32
C240: 06 CE 05 D0 4C 13 C3 A9 B7
C248: 03 85 FC 4C BC C7 C9 01 69
C250: D0 28 AD 04 D0 C9 5F B0 A5
C258: 07 A9 02 85 FC 4C DA C6 7B
C260: CE 04 D0 CE 04 D0 AD 05 5A
C268: D0 C9 37 90 06 CE 05 D0 75
C270: 4C 13 C3 A9 04 85 FC 4C 10
C278: BC C7 C9 02 D0 28 AD 04 73
C280: D0 C9 F2 90 07 A9 01 85 D5
C288: FC 4C DA C6 EE 04 D0 EE 26
C290: 04 D0 AD 05 D0 C9 37 90 7A
C298: 06 CE 05 D0 4C 13 C3 A9 10
C2A0: 05 85 FC 4C BC C7 C9 03 C5
C2A8: D0 14 AD 05 D0 C9 E0 B0 6C
C2B0: 06 EE 05 D0 4C 13 C3 A9 48
C2B8: 00 85 FC 4C D7 C7 C9 04 F4
C2C0: D0 28 AD 04 D0 C9 5F B0 16
C2C8: 07 A9 05 85 FC 4C DA C6 EE
C2D0: CE 04 D0 CE 04 D0 AD 05 CA
C2D8: D0 C9 E0 B0 06 EE 05 D0 CF
C2E0: 4C 13 C3 A9 01 85 FC 4C 7D
C2E8: D7 C7 C9 05 D0 25 AD 04 FE
C2F0: D0 C9 F2 90 07 A9 04 85 49
C2F8: FC 4C DA C6 EE 04 D0 EE 96
C300: 04 D0 AD 05 D0 C9 E0 90 93
C308: 07 A9 02 85 FC 4C D7 C7 29
C310: EE 05 D0 AD 01 DC 29 10 99
C318: D0 08 A9 06 8D 27 D0 4C 72
C320: 27 C3 A9 02 8D 27 D0 AD E9
C328: 01 DC 29 0F 8D A7 02 AD 23
C330: 01 DC 29 10 18 6D A7 02 76
C338: C9 1B D0 06 20 BC C3 4C E0
C340: EE C3 C9 17 D0 06 20 CA 95
C348: C3 4C EE C3 C9 1D D0 06 C8
C350: 20 D8 C3 4C EE C3 C9 1E F3
C358: D0 06 20 E3 C3 4C EE C3 F5
C360: C9 1A D0 09 20 E3 C3 20 06
C368: BC C3 4C EE C3 C9 16 D0 98
C370: 09 20 E3 C3 20 CA C3 4C 3C
C378: EE C3 C9 19 D0 09 20 D8 E0
C380: C3 20 BC C3 4C EE C3 C9 AD
C388: 15 D0 09 20 D8 C3 20 CA 1F
C390: C3 4C EE C3 AC B3 02 C0 76
C398: 01 D0 F6 C9 0D D0 07 A9 B9
C3A0: 03 85 FC 4C F4 C6 C9 0B 03
C3A8: D0 07 A9 04 85 FC 4C F4 F1
C3B0: C6 C9 07 D0 DC A9 05 85 2A
C3B8: FC 4C F4 C6 AD 00 D0 C9 06
C3C0: 62 90 06 CE 00 D0 CE 00 28
C3C8: D0 60 AD 00 D0 C9 EF B0 E2
C3D0: F8 EE 00 D0 EE 00 D0 60 A9
C3D8: AD 01 D0 C9 7E B0 EA EE 2B
C3E0: 01 D0 60 AD 01 D0 C9 54 B0
C3E8: 90 DF CE 01 D0 60 AD 00 08
C3F0: DC 29 10 D0 08 A9 06 8D 1D
C3F8: 28 D0 4C 02 C4 A9 04 8D 40
C400: 28 D0 AD 00 DC 29 0F 8D 49
C408: A9 02 AD 00 DC 29 10 18 8F
C410: 6D A9 02 C9 1B D0 06 20 05
C418: 9A C4 4C CC C4 C9 17 D0 07
C420: 06 20 A8 C4 4C CC C4 C9 5B
C428: 1D D0 06 20 B6 C4 4C CC D0
C430: C4 C9 1E D0 06 20 C1 C4 5A
C438: 4C CC C4 C9 1A D0 09 20 F3
C440: C1 C4 20 9A C4 4C CC C4 24
C448: C9 16 D0 09 20 C1 C4 20 C8
C450: A8 C4 4C CC C4 C9 19 D0 4F
C458: 09 20 B6 C4 20 9A C4 4C C8
C460: CC C4 C9 15 D0 09 20 B6 81
C468: C4 20 A8 C4 4C CC C4 AC 45
C470: B3 02 C0 02 D0 21 C9 0E B2
```

```
C478: D0 07 A9 00 85 FC 4C F4 BD
C480: C6 C9 0B D0 07 A9 01 85 24
C488: FC 4C F4 C6 C9 07 D0 07 36
C490: A9 02 85 FC 4C F4 C6 4C 13
C498: CC C4 AD 02 D0 C9 62 90 67
C4A0: 06 CE 02 D0 CE 02 D0 60 4A
C4A8: AD 02 D0 C9 EF B0 F8 EE 7B
C4B0: 02 D0 EE 02 D0 60 AD 03 56
C4B8: D0 C9 C3 B0 EA EE 03 D0 75
C4C0: 60 AD 03 D0 C9 99 90 DF 76
C4C8: CE 03 D0 60 AD F0 02 C9 36
C4D0: 01 D0 0A AD B3 02 C9 01 DA
C4D8: D0 0D 4C 8B C5 4C E6 C5 4D
C4E0: A5 FE 85 FD 4C 6E C5 A5 2F
C4E8: FD F0 F5 AD B3 02 D0 2C 2E
C4F0: AD 00 D0 CD 04 D0 F0 43 46
C4F8: 90 11 CE 00 D0 AD 00 D0 B8
C500: CD 04 D0 F0 36 CE 00 D0 69
C508: 4C 3B C5 EE 00 D0 AD 00 C2
C510: D0 CD 04 D0 B0 25 EE 00 48
C518: D0 4C 3B C5 AD 01 D0 C9 7F
C520: 79 B0 03 EE 01 D0 AD 00 BB
C528: D0 C9 AA F0 3F 90 06 CE 03
C530: 00 D0 4C 6C C5 EE 00 D0 3F
C538: 4C 6C C5 AD 05 D0 C9 8A 8E
C540: B0 0D AD 01 D0 C9 54 90 2C
C548: 23 CE 01 D0 4C 6C C5 A5 30
C550: FC C9 AA F0 0D AD 01 D0 3F
C558: C9 6C B0 10 EE 01 D0 4C 5C
C560: 6C C5 AD 01 D0 C9 7E B0 0B
C568: 03 EE 01 D0 C6 FD AD B3 52
C570: 02 D0 12 AD 1E D0 C9 05 C0
C578: D0 0E A9 01 8D B3 02 8D D2
C580: AD 02 4C F8 C5 4C E6 C5 34
C588: 4C 09 C6 AD 01 D0 C9 60 4E
C590: 90 03 CE 01 D0 AD AF 02 24
C598: F0 10 AD 00 D0 C9 F0 B0 83
C5A0: 06 EE 00 D0 4C BD C5 CE 05
C5A8: AF 02 AD 00 D0 C9 62 90 95
C5B0: 06 CE 00 D0 4C BD C5 EE 15
C5B8: AF 02 4C A1 C5 AD 00 D0 9C
C5C0: 8D 04 D0 AD 01 D0 8D 05 35
C5C8: D0 A5 A2 C9 60 B0 14 AD 7E
C5D0: 04 DC C9 A0 B0 07 C9 60 FD
C5D8: 90 06 4C 9F C3 4C AA C3 D9
C5E0: 4C B5 C3 4C 40 C7 AD B3 5C
C5E8: 02 C9 01 F0 0B C9 02 F0 6E
C5F0: 1C AD 1E D0 C9 05 D0 11 5A
C5F8: AD 00 D0 8D 04 D0 AD 01 88
C600: D0 8D 05 D0 A9 01 4C 1B 46
C608: C6 C9 06 D0 17 AD 02 D0 07
C610: 8D 04 D0 AD 03 D0 8D 05 86
C618: D0 A9 02 8D AD 02 8D B3 13
C620: 02 4C 40 C7 AE BE 02 E0 C6
C628: 00 F0 03 4C 59 C8 C9 0C 60
C630: D0 11 AD 15 D0 38 E9 08 CF
C638: 8D 15 D0 A9 01 8D BE 02 A4
C640: 4C 55 C6 C9 14 D0 1B AD 20
C648: 15 D0 38 E9 10 8D 15 D0 D3
C650: A9 02 8D BE 02 A9 03 85 7C
C658: FC 20 F2 C7 20 F2 C7 4C 57
C660: 93 C6 C9 24 D0 11 AD 15 4D
C668: D0 38 E9 20 8D 15 D0 A9 98
C670: 03 8D BE 02 4C 89 C6 C9 28
C678: 44 D0 2F AD 15 D0 38 E9 72
C680: 40 8D 15 D0 A9 04 8D BE 2
C688: 02 A9 00 85 FC 20 15 C8 B4
C690: 20 15 C8 20 38 C8 A9 15 6E
C698: 8D 04 D4 A9 F0 8D 06 D4 02
C6A0: A9 40 8D 01 D4 A9 0D 8D 32
C6A8: C0 02 4C 9E C8 20 38 C8 40
C6B0: A9 15 8D 04 D4 A9 F0 8D FD
C6B8: 06 D4 A9 18 8D 01 D4 A9 62
C6C0: 0D 8D C0 02 4C 13 C3 20 61
C6C8: 38 C8 A9 21 8D 04 D4 A9 A4
C6D0: F0 8D 06 D4 A9 02 8D 01 64
C6D8: D4 60 20 38 C8 A9 21 8D 87
C6E0: 04 D4 A9 F0 8D 06 D4 A9 66
C6E8: 06 8D 01 D4 A9 0D 8D C0 57
C6F0: 02 4C 13 C3 20 38 C8 A9 E0
C6F8: 81 8D 04 D4 A9 F0 8D 06 0F
C700: D4 A9 05 8D 01 D4 A9 0D 9D
C708: 8D C0 02 AD B3 02 C9 01 86
C710: D0 0C AD 01 D0 18 69 18 06
C718: 8D 05 D0 4C 27 C7 AD 03 67
C720: D0 38 E9 18 8D 05 D0 A9 38
C728: 00 8D AD 02 8D B3 02 8D 36
C730: BB 02 A9 33 8D 03 06 8D EE
C738: 53 06 AD 1E D0 4C 9E C8 E1
C740: AD BB 02 D0 15 A9 FF 8D C8
C748: C8 02 A9 03 8D C9 02 EE 08
C750: BB 02 20 C7 C6 A9 10 8D 04
C758: C0 02 AD C8 02 D0 57 CE 8A
C760: C9 02 AD C9 02 18 69 30 57
C768: AE B3 02 E0 01 D0 06 8D 13
C770: 03 06 4C 78 C7 8D 53 06 EC
C778: AD C9 02 D0 39 AD B3 02 5F
C780: C9 01 D0 06 20 F2 C7 4C 49
C788: 8D C7 20 15 C8 20 C7 C6 8A
C790: A9 FF 85 FB 20 4A C8 20 0F
C798: 38 C8 20 4A C8 AD C2 02 3F
C7A0: F0 03 4C DE C8 A9 00 8D BF
C7A8: AD 02 8D B3 02 8D BB 02 E6
C7B0: 8D CF 02 4C DE C1 CE C8 94
C7B8: 02 4C 9E C8 AD 34 06 18 6E
C7C0: 69 05 C9 35 D0 06 8D 34 C6
C7C8: 06 4C AD C6 A9 30 8D 34 2B
C7D0: 06 20 F2 C7 4C AD C6 AD 20
C7D8: E4 05 18 69 05 C9 35 D0 19
C7E0: 06 8D E4 05 4C AD C6 A9 C8
C7E8: 30 8D E4 05 20 15 C8 4C DA
C7F0: AD C6 AD 1E D0 AD 33 06 E8
C7F8: C9 39 B0 04 EE 33 06 60 39
C800: A9 30 8D 33 06 AD 32 06 86
C808: C9 31 90 05 A9 02 8D C2 94
C810: 02 EE 32 06 60 AD 1E D0 36
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 97 and 98 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
C818: AD E3 05 C9 39 B0 04 EE 55
C820: E3 05 60 A9 30 8D E3 05 B9
C828: AD E2 05 C9 31 90 05 A9 F7
C830: 01 8D C2 02 EE E2 05 60 BA
C838: A0 00 98 99 00 D4 C0 17 B7
C840: B0 04 C8 4C 3B C8 8D C0 5C
C848: 02 60 A2 00 A0 00 C8 C0 77
C850: FF D0 FB E8 E4 FB D0 F4 AC
C858: 60 AD 05 D0 C9 A0 B0 3E 95
C860: C9 76 90 3A E0 01 D0 0B 29
C868: AD 15 D0 09 10 8D 15 D0 88
C870: 4C 99 C8 E0 02 D0 0B AD 8B
C878: 15 D0 09 08 8D 15 D0 4C 2F
C880: 99 C8 E0 03 D0 0B AD 15 65
C888: D0 09 40 8D 15 D0 4C 99 FB
C890: C8 AD 15 D0 09 20 8D 15 B8
C898: D0 A9 00 8D BE 02 20 4A CB
C8A0: C8 A0 00 C8 C0 80 D0 FB E0
C8A8: 20 E1 FF D0 19 20 38 C8 B5
C8B0: 20 77 C9 20 E4 FF F0 FB 04
C8B8: C9 4E D0 03 4C 00 C0 C9 7B
C8C0: 43 D0 F0 20 87 C9 AD C2 A7
C8C8: 02 D0 13 AD C0 02 F0 08 18
C8D0: CE C0 02 AD C0 02 D0 03 A6
C8D8: 20 38 C8 4C 31 C2 C9 01 05
C8E0: D0 13 A0 00 B9 E2 05 18 1F
C8E8: 69 80 99 E2 05 C8 C0 03 E0
C8F0: 90 F2 4C 05 C9 A0 00 B9 E9
C8F8: 32 06 18 69 80 99 32 06 05
C900: C8 C0 03 90 F2 A2 27 86 60
C908: FB 20 4A C8 84 FB 20 38 10
C910: C8 20 4A C8 A9 15 8D 04 5C
C918: D4 A9 F0 8D 06 D4 A0 00 90
C920: 8C F8 02 8C F9 02 AC F9 D6
C928: 02 B9 FC C9 8D 01 D4 B9 C7
C930: FD C9 85 FB 20 4A C8 EE 9B
C938: F9 02 EE F9 02 AD F9 02 C8
C940: C9 0B 90 E2 A9 00 8D 01 C0
C948: D4 20 4A C8 EE F8 02 AD E7
C950: F8 02 C9 03 B0 05 A0 00 6E
C958: 4C 23 C9 20 38 C8 20 77 4A
C960: C9 20 E4 FF F0 FB C9 4E 34
C968: D0 03 4C 00 C0 C9 43 D0 27
C970: F0 20 87 C9 4C 81 C0 A0 02
C978: 00 84 C6 B9 E4 C9 20 D2 1F
C980: FF C8 C0 0C 90 F5 60 A0 9D
C988: 00 B9 F0 C9 20 D2 FF C8 B8
C990: C0 0C 90 F5 60 99 93 0D 7E
C998: 0D 0D 20 20 4C 45 56 45 20
C9A0: 4C 20 20 20 20 28 31 2D F3
C9A8: 32 2D 33 29 3F 20 0D 0D DD
C9B0: 20 20 50 4C 41 59 45 52 BF
C9B8: 53 20 20 20 20 28 31 2D 13
C9C0: 32 29 3F 20 0D 0D 20 20 D5
C9C8: 43 4F 4D 50 55 54 45 52 3A
C9D0: 20 28 31 2D 32 2D 33 29 33
C9D8: 3F 20 13 03 0F 12 05 14 88
C9E0: 09 0D 05 20 13 4E 3D 4E 09
C9E8: 45 57 0D 43 3D 43 4F 4E F3
C9F0: 13 20 20 20 20 20 0D 20 D1
C9F8: 20 20 20 20 0C 60 0E 60 54
CA00: 10 60 13 C0 10 60 13 FF C7
CA08: 0E 00 70 0E 00 70 03 00 08
CA10: C0 01 FF 80 03 FF C0 0F 25
CA18: FF F0 1F FF F8 3F FF FC 5D
CA20: 7F FF FE FF FF FF FF FF 9E
CA28: FF E7 FF E7 E3 FF C7 70 14
CA30: 00 07 3F E0 0E 1F 87 FC 09
CA38: 0E 01 F8 07 80 70 03 E1 1D
CA40: E0 00 07 C0 00 00 00 00 E8
CA48: 00 00 00 03 E0 00 07 87 BA
CA50: C0 0E 01 E0 1F 80 70 3F 50
CA58: E1 F8 70 07 FC E0 00 0E 96
CA60: E1 C3 87 E1 C3 87 E3 00 9E
CA68: C7 F1 FF 8F FF FF FF FF B1
CA70: FF FF 7F FF FE 3F FF FC 2B
CA78: 1F FF F8 0F FF F0 03 FF 93
CA80: C0 00 FF 00 00 00 00 00 41
CA88: 00 00 00 00 00 00 00 00 88
CA90: 00 00 00 00 00 3C 00 00 CC
CA98: 7E 00 03 FF C0 03 FF C0 9E
CAA0: 0F FF F0 0F FF F0 3F 3C 1C
CAA8: FC 3F 3C FC 0F FF F0 0F 2D
CAB0: FF F0 03 FF C0 03 FF C0 29
CAB8: 00 7E 00 00 3C 00 00 00 73
CAC0: 00 00 00 00 00 00 00 00 C0
CAC8: 00 FE 00 01 FF 00 73 FF 3C
CAD0: 80 C3 39 80 C3 39 80 E3 30
CAD8: FF 80 71 FF 00 18 FE 00 E1
CAE0: 0C 7C 78 06 7C CE 03 FF 36
CAE8: 87 01 FF 03 03 FC 03 0F 86
CAF0: F8 03 38 30 40 60 30 60 86
CAF8: 60 18 30 30 0C 18 18 06 14
CB00: 18 0C 03 18 07 81 F0 00 B8
```

CHARACTER DUMP
FROM PAGE 56

```
·1 REM ********************************     NK
·2 REM **                            **     DP
·3 REM **      CHARACTER DUMP        **     PC
·4 REM **      BY JIM SANDERS        **     BG
·5 REM **    FOR AHOY! MAGAZINE      **     PK
·6 REM **                            **     DP
·7 REM ********************************     NK
·8 POKE 53280,12:POKE53281,12:PRINT"[HOME
 ][BLUE]"                                   AI
·10 REM*DATA LOADER FOR DUMP 52992          ID
·20 PRINT"[CLEAR]":CS=0                     BG
·30 FOR I=52992 TO 53140:READ A:POKE I,A    ID
·40 CS=CS+A:NEXTI                           MM
```

```
•45 PRINTCS                                  DG
•50 IFCS<>18597 THEN PRINT"[4"*"]ERROR IN
  DATA STATEMENTS[4"*"]":END                 OM
•55 PRINT"TO ACTIVATE F7 KEY SYS 52992"     PJ
•60 PRINT"TO CHANGE TO F1 POKE 53024,4"     MP
•65 PRINT"TO CHANGE TO F3 POKE 53024,5"     FM
•70 PRINT"TO CHANGE TO F5 POKE 53024,6"     EF
•75 PRINT"TO CHANGE TO F7 POKE 53024,3"     IO
•80 PRINT"TYPE 'NEW' <RETURN> AFTER POKE"   IA
•85 PRINT"HIT RUN/STOP TO DISABLE"          OE
•90 PRINT"SYS 52992 THEN PRESS F7":PRINT"
 TO PRINT THESE INSTRUCTIONS"               II
•95 PRINT"CHARACTER DUMP MAY BE ACTIVATED
  FROM":PRINT"PROGRAM BY LINE ENTRY:SYS 5
 3034"                                      FA
•100 NEW                                    KA
•52992 DATA120,173,20,3,141,169,2,173       KC
•53000 DATA21,3,141,170,2,169,29,141        MG
•53008 DATA20,3,169,207,141,21,3,169        EF
•53016 DATA0,133,254,88,96,165,197,201      PH
•53024 DATA3,208,4,165,254,240,3,108        NJ
•53032 DATA169,2,169,255,133,254,169,4      NI
•53040 DATA133,186,169,126,133,184,169,0    DI
•53048 DATA160,4,133,113,132,114,133,183    JL
•53056 DATA133,185,32,192,255,166,184,32    PC
•53064 DATA201,255,162,25,169,13,32,210     IG
•53072 DATA255,32,225,255,240,46,160,0      MD
•53080 DATA177,113,133,103,41,63,6,103      NL
•53088 DATA36,103,16,2,9,128,112,2          DI
•53096 DATA9,64,32,210,255,200,192,40       EP
•53104 DATA208,230,152,24,101,113,133,113   GO
•53112 DATA144,2,230,114,202,208,205,169    NA
•53120 DATA13,32,210,255,32,204,255,169     HA
•53128 DATA8,32,195,255,169,0,133,254       KK
•53136 DATA76,188,254,96,255                BB
```

HEAD TO HEAD
FROM PAGE 72

```
•1 REM ********************************     DN
•2 REM *                              *     OO
•3 REM *    'HEAD TO HEAD' (C) 1985   *     FL
•4 REM *        BY    DOUG SMOAK      *     BM
•5 REM *              303 HEYWARD ST. *     GF
•6 REM *              COLUMBIA, S.C.  *     GC
•7 REM *              29201           *     OG
•8 REM *                              *     OO
•9 REM ********************************     DN
•10 DIMA(3),X(4),C(4)                       FB
•20 A(0)=2:A(2)=-80:A(1)=-2:A(3)=80         PJ
•30 X(0)=0:X(1)=-40:X(2)=40:X(3)=-1:X(4)=
 1                                          DN
•40 C(0)=0:C(1)=30:C(2)=22:C(3)=60:C(4)=6
 2                                          LJ
•50 BO=53280:BG=BO+1:SC=1024:WL=160:HL=46   JP
•60 SID=54272:CD=SID                        AM
•70 DN$="[HOME][14"[DOWN]"]"                EI
•80 DS$=DN$+"[4"[DOWN]"][13"[RIGHT]"]"      MG
•90 GOSUB880:POKEBO,0:POKEBG,0:POKE646,14   IK
•100 GOSUB1600                              FM
•110 FORI=49152TO49152+213                  LA
•120 READA:POKEI,A:NEXT                     BD
•130 GOSUB1130                              FM
•140 PRINT"[CLEAR][c 8]"DN$"BONUS FEATURE
  [RVSON]Y/N[RVSOFF]";                      DA
•150 INPUTBS$                               EF
•160 PRINT"[CLEAR]"DN$"HOW MANY ROUNDS";    ON
•170 INPUTRD$:RD=VAL(RD$)                   NH
•180 IFRD=0THENRD=1                         CP
•190 :                                      DI
•200 REM ** MAIN LOOP **                    JI
•210 :                                      DI
•220 FORII=1TORD                            EF
•230 POKEBO,0:POKEBG,0                      DN
•240 M1=1905:M2=1925:C1=30:C2=C1:TP=0       LI
•250 POKE198,0                              KB
•260 PRINT"[CLEAR]"DN$"WHAT NUMBER MAZE";   PN
•270 INPUTMZ$:MZ=VAL(MZ$):MZ$=""            OP
•280 PRINT"[CLEAR]":IFMZ=0THENMZ=RND(1)     CH
•290 FORI=0TO22                             MF
•300 PRINT"[RED][RVSON][19" "][RIGHT][19"
 "][RVSOFF][RIGHT]";                        LL
•310 NEXT                                   IA
•320 A9=SC+81:GOSUB670:MZ=0                 CK
•330 IFBS$<>"Y"THEN390                      JG
•340 X=INT(RND(1)*5)+2                      OA
•350 Y=INT(RND(1)*8)+2                      EC
•360 FORI=SC+X*40+YTOSC+X*40+19             OA
•370 IFPEEK(I)=160THENPOKEI,86:POKEI+20,8
 6:POKEI+CD,7:POKEI+20+CD,7:GOTO390         LH
•380 NEXT                                   IA
•390 TI$="[6"0"]"                           NI
•400 POKEM1,C1:POKEM2,C2                    KG
•410 O1=M1:O2=M2                            LF
•420 POKE49301,255:SYS49223                 IG
•430 PRINT"[HOME][YELLOW]ROUND"IITAB(20)"
 SEC"10-VAL(RIGHT$(TI$,2))"[LEFT] "         GE
•440 IFVAL(RIGHT$(TI$,2))<10THEN430         GN
•450 SYS49152:J1=PEEK(253)                  KM
•460 M1=O1+X(J1):W1=PEEK(M1)                PO
•470 IFW1=C1ORW1=WLTHEN520                  OL
•480 IFW1=HLTHENTP=TP+1:P1=P1+10:POKEM1+2
 0,32                                       AG
•490 IFW1=86THENP1=P1+500:POKEM1+20,32      EJ
•500 POKEO1,32:POKEM1,C(J1)                 FP
•510 O1=M1:C1=C(J1):GOTO530                 IP
•520 M1=O1                                  LA
•530 SYS49152:J2=PEEK(254)                  LA
•540 M2=O2+X(J2):W2=PEEK(M2)                MB
•550 IFW2=C2ORW2=WLTHEN600                  OL
•560 IFW2=HLTHENTP=TP+1:P2=P2+10:POKEM2-2
 0,32                                       AJ
•570 IFW2=86THENP2=P2+500:POKEM2-20,32      DI
•580 POKEO2,32:POKEM2,C(J2)                 BC
•590 O2=M2:C2=C(J2):GOTO610                 JE
```

```
·600 M2=02                                      LE
·610 PRINT"[HOME][YELLOW]SCORE"P1TAB(20)"
 SCORE"P2                                       OK
·620 IFTP>=196THENSYS49287:GOSUB880:GOSUB
 1090:NEXTII:GOTO920                            JA
·630 GOTO450                                    CN
·640 :                                          DI
·650 REM *** CREATE PLAYFIELD ***               OH
·660 :                                          DI
·670 ZM=RND(-MZ)                                KN
·680 J=INT(RND(1)*4):X3=J                       AB
·690 B=A9+A(J)                                  AG
·700 IFPEEK(B)=WLTHENGOSUB800:GOTO680           GE
·710 J=(J+1)*-(J<3):IFJ<>X3THEN690              HI
·720 J=PEEK(A9)                                 JM
·730 POKEA9,HL:POKEA9+CD,1                      LB
·740 POKEA9+20,HL:POKEA9+CD+20,1                AM
·750 IFJ<4THENA9=A9-A(J):GOTO680                OI
·760 RETURN                                     IM
·770 :                                          DI
·780 REM *** PLAYFIELD SUB ***                  BM
·790 :                                          DI
·800 POKEB,J:POKEA9+A(J)/2,HL                   MP
·810 POKEA9+CD+A(J)/2,1                         CD
·820 POKEB+20,J:POKE(A9+A(J)/2)+20,HL           OO
·830 POKE20+A9+CD+A(J)/2,1:A9=B                 OJ
·840 RETURN                                     IM
·850 :                                          DI
·860 REM ** SETUP SOUND CHIP **                 GM
·870 :                                          DI
·880 FORI=0TO24:POKESI+I,0:NEXT:POKESI+24
 ,15:POKESI+6,240:RETURN                        JL
·890 :                                          DI
·900 REM *** GAME OVER ***                      PO
·910 :                                          DI
·920 IFP1>P2THENPL$="PLAYER ONE, THAT'S Y
 OU ON THE LEFT!":GOTO950                       MD
·930 IFP2>P1THENPL$="PLAYER TWO, THAT'S Y
 OU ON THE RIGHT!":GOTO950                      KM
·940 IFP2=P1THENPL$="[7" "]NOBODY, IT'S A
  TIE[3"!"]??"                                  KJ
·950 PRINT"[CLEAR][DOWN][RIGHT][YELLOW]TH
 E FINAL SCORES ARE:[DOWN]"                     KH
·960 PRINT"[DOWN][RIGHT]PLAYER ONE:[WHITE
 ]"P1                                           JB
·970 PRINT"[YELLOW][DOWN][RIGHT]PLAYER TW
 O:[WHITE]"P2                                   IO
·980 PRINT"[YELLOW][3"[DOWN]"][RIGHT]THE
 WINNER IS:"                                    KL
·990 PRINT"[DOWN][DOWN][RIGHT][RIGHT][WHI
 TE]"PL$                                        AA
·1000 PRINT"[YELLOW][5"[DOWN]"][12" "]PLA
 Y AGAIN  [RVSON]Y/N[RVSOFF] ?";                BG
·1010 FORI=1TO1000:NEXT:POKE198,0               JN
·1020 GETA$                                     AL
·1030 IFA$="N"THENPRINT"[CLEAR]";:END           FH
·1040 IFA$<>"Y"THEN1020                         HB
·1050 P1=0:P2=0:RD=0:II=0:GOTO140               PM
·1060 :                                         DI
·1070 REM  *** CHECK FOR END ***                MD
·1080 :                                         DI
·1090 IFRD>0ANDII=RDTHENRETURN                  MG
·1100 :                                         DI
·1110 REM  *** WAIT FOR INPUT ***               CO
·1120 :                                         DI
·1130 PRINT"[c 1][HOME][24"[DOWN]"][8"[RI
 GHT]"]PRESS RETURN TO CONTINUE";               CF
·1140 CL=RND(0)*15+1                            GH
·1150 IFRD=0THENPOKE646,CL:GOSUB1600:PRIN
 TDS$"BY DOUG SMOAK "                           NJ
·1160 IF(PEEK(197)<>1)THEN1140                  JM
·1170 POKE198,0:RETURN                          HE
·1180 :                                         DI
·1190 REM ** M.L. J-STICK DATA **               EA
·1200 :                                         DI
·1210 DATA169,0,133,254,173,0,220               LD
·1220 DATA74,176,4,160,1,132,254,74             AA
·1230 DATA176,4,160,2,132,254,74,176            NJ
·1240 DATA4,160,3,132,254,74,176                IE
·1250 DATA4,160,4,132,254,169,0,133             FI
·1260 DATA253,173,1,220,74,176,4                NL
·1270 DATA160,1,132,253,74,176,4                LL
·1280 DATA160,2,132,253,74,176,4,160            OF
·1290 DATA3,132,253,74,176,4,160                AD
·1300 DATA4,132,253,96                          MK
·1310 :                                         DI
·1320 REM ** INTERRUPT MUSIC DATA **            ME
·1330 :                                         DI
·1340 DATA120,169,84,141,20,3,169               GA
·1350 DATA192,141,21,3,88,96,173,148            MD
·1360 DATA192,208,40,174,149,192                NE
·1370 DATA232,224,64,208,2,162,0,189            HM
·1380 DATA150,192,141,0,212,232,189             ON
·1390 DATA150,192,141,1,212,232,189             LM
·1400 DATA150,192,141,4,212,232,189             HB
·1410 DATA150,192,141,148,192,142,149           EA
·1420 DATA192,206,148,192,76,49,234             NN
·1430 DATA120,169,49,141,20,3,169               IJ
·1440 DATA234,141,21,3,88,96                    AK
·1450 DATA 0,255                                HJ
·1460 :                                         DI
·1470 REM ** MUSIC NOTES DATA **                HK
·1480 :                                         DI
·1490 DATA 96,22,17,15,195,16,17,15             AC
·1500 DATA 24,14,17,15,195,16,17,15             IE
·1510 DATA 96,22,17,15,195,16,17,15             AC
·1520 DATA 24,14,17,15,195,16,17,15             IE
·1530 DATA 239,14,17,15,195,16,17,15            HM
·1540 DATA 239,14,17,15,195,16,17,15            HM
·1550 DATA 239,14,17,15,195,16,17,15            HM
·1560 DATA 209,18,17,15,31,21,17,15             AC
·1570 :                                         DI
·1580 REM *** TITLE PAGE ***                    DL
·1590 :                                         DI
·1600 PRINT"[HOME][39" "]";                     HM
·1610 PRINT"[7" "][RVSON][c A][26"[s *]"]
```

```
 [c S][RVSOFF][5" "]";                        PH
·1620 PRINT"[7" "][RVSON][s -][RVSOFF][26
 " "][RVSON][s -][RVSOFF][5" "]";             BP
·1630 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON][3" "][RVSOFF] [RVSON][sEP] [c *][RV
 SOFF] [RVSON]  [c *][RVSOFF][6" "][RVSON
 ][s -][RVSOFF][5" "]";                       JB
·1640 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON] [RVSOFF][3" "][RVSON] [RVSOFF] [RVS
 ON] [RVSOFF] [RVSON] [RVSOFF] [RVSON] [R
 VSOFF][6" "][RVSON][s -][RVSOFF][5" "]";     MN
·1650 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [c I] [RVSOFF] [RVSON] [c I][
 RVSOFF]  [RVSON] [c I] [RVSOFF] [RVSON]
 [RVSOFF] [RVSON] [RVSOFF][6" "][RVSON][s
 -][RVSOFF][5" "]";                           LE
·1660 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON] [RVSOFF][c I][c I] [RVSON] [RVSOFF]
 [RVSON] [RVSOFF] [RVSON] [RVSOFF][c I][
 RVSON] [RVSOFF][6" "][RVSON][s -][RVSOFF
 ][5" "]";                                    FG
·1670 PRINT"[7" "][RVSON][s -][RVSOFF][26
 " "][RVSON][s -][RVSOFF][5" "]";             BP
·1680 PRINT"[7" "][RVSON][s -][RVSOFF][9"
 "][RVSON][3" "][RVSOFF] [RVSON][3" "][R
 VSOFF][10" "][RVSON][s -][RVSOFF][5" "]"
 ;                                            KL
·1690 PRINT"[7" "][RVSON][s -][RVSOFF][10
 " "][RVSON] [RVSOFF]  [RVSON] [RVSOFF] [
 RVSON] [RVSOFF][10" "][RVSON][s -][RVSOF
 F][5" "]";                                   KF
·1700 PRINT"[7" "][RVSON][s -][RVSOFF][10
 " "][RVSON] [RVSOFF]  [RVSON] [RVSOFF] [
 RVSON] [RVSOFF][10" "][RVSON][s -][RVSOF
 F][5" "]";                                   KF
·1710 PRINT"[7" "][RVSON][s -][RVSOFF][10
 " "][RVSON] [RVSOFF]  [RVSON] [RVSOFF][c
 I][RVSON] [RVSOFF][10" "][RVSON][s -][R
 VSOFF][5" "]";                               NG
·1720 PRINT"[7" "][RVSON][s -][RVSOFF][26
 " "][RVSON][s -][RVSOFF][5" "]";             BP
·1730 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON][3" "][RVSOFF] [RVSON][sEP] [c *][RV
 SOFF] [RVSON]  [c *][RVSOFF][6" "][RVSON
 ][s -][RVSOFF][5" "]";                       JB
·1740 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON] [RVSOFF][3" "][RVSON] [RVSOFF] [RVS
 ON] [RVSOFF] [RVSON] [RVSOFF] [RVSON] [R
 VSOFF][6" "][RVSON][s -][RVSOFF][5" "]";     MN
·1750 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [c I] [RVSOFF] [RVSON] [c I][
 RVSOFF]  [RVSON] [c I] [RVSOFF] [RVSON]
 [RVSOFF] [RVSON] [RVSOFF][6" "][RVSON][s
 -][RVSOFF][5" "]";                           LE
·1760 PRINT"[7" "][RVSON][s -][RVSOFF][5"
 "][RVSON] [RVSOFF] [RVSON] [RVSOFF] [RV
 SON] [RVSOFF][c I][c I] [RVSON] [RVSOFF]
 [RVSON] [RVSOFF] [RVSON] [RVSOFF][c I][
 RVSON] [RVSOFF][6" "][RVSON][s -][RVSOFF
 ][5" "]";                                    FG
·1770 PRINT"[7" "][RVSON][s -][RVSOFF][26
 " "][RVSON][s -][RVSOFF][5" "]";             BP
·1780 PRINT"[7" "][RVSON][s -][RVSOFF][6"
 "]BY DOUG SMOAK[7" "][RVSON][s -][RVSOF
 F][5" "]";                                   LH
·1790 PRINT"[7" "][RVSON][s -][RVSOFF][26
 " "][RVSON][s -][RVSOFF][5" "]";             BP
·1800 PRINT"[7" "][RVSON][c Z][26"[s *]"]
 [c X][RVSOFF][5" "]";                        PH
·1810 PRINT"[40" "]";                         JJ
·1820 PRINT"[40" "]";                         JJ
·1830 PRINT"[40" "][HOME]";                   HM
·1840 POKE646,RND(0)*15+1                     KG
·1850 RETURN                                  IM
```


...COMING IN THE MARCH *AHOY!* (ON SALE FEBRUARY 4)...

CHARACTER DEVELOPMENT
ML SHORTCUTS TO CUSTOMIZING YOUR COMMODORE

128 DELIGHTS
THE RUPERT REPORT SWITCHES TO 128 MODE!

PROGRAMMING EDUCATIONAL GAMES
MAKING LEARNING FUN FOR ITS OWN SAKE

DON'T MISS THE BOAT!



SUBSCRIBE TO AHOY!

☐ Send me twelve issues of AHOY! for $19.95 ($26.95 Canada & elsewhere). I understand that I will be billed later.

☐ Send me twenty-four issues of AHOY! for $37.95 ($49.95 Canada and elsewhere). I understand that I will be billed later.

NAME ____________________

ADDRESS ____________________

CITY ____________________

STATE __________ ZIP __________

DON'T MISS THE BOAT!



SUBSCRIBE TO THE AHOY! DISK MAGAZINE

☐ Send me 12 issues of the AHOY! Disk Magazine for $88.00 ($115.00 Canada and elsewhere). I understand that I will be billed later.

☐ Send me 24 issues of the AHOY! Disk Magazine for $165.00 ($214.00 Canada and elsewhere). I understand that I will be billed later.

NAME ____________________

ADDRESS ____________________

CITY ____________________

STATE __________ ZIP __________

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 65 MT. MORRIS, ILL.

POSTAGE WILL BE PAID BY ADDRESSEE

Ahoy!

P.O. BOX 341
MT. MORRIS, ILL. 61054

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 65 MT. MORRIS, ILL.

POSTAGE WILL BE PAID BY ADDRESSEE

Ahoy!

P.O. BOX 341
MT. MORRIS, ILL. 61054

DON'T MISS THE BOAT!



SUBSCRIBE TO AHOY!

☐ Send me twelve issues of AHOY! for $19.95 ($26.95 Canada & elsewhere). I understand that I will be billed later.

☐ Send me twenty-four issues of AHOY! for $37.95 ($49.95 Canada & elsewhere). I understand that I will be billed later.

NAME ______________________

ADDRESS ______________________

CITY ______________________

STATE __________ ZIP __________

March 1986 *Ahoy!*/Void after March 11, 1986

READER SERVICE CARD

To request additional information on any product advertised or otherwise pictured in this issue of *Ahoy!* that is accompanied by a reader service number, circle the corresponding number below and mail this card. We will promptly forward your request to the designated companies.

101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130
131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190
191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	210
211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300

A. Please check whether you are...

1. ☐ male 2. ☐ female

B. What is your age?

1. ☐ under 18 2. ☐ 18-24 3. ☐ 25-34
4. ☐ 35-44 5. ☐ 45-54 6. ☐ 55-64 7. ☐ 65+

C. Education level completed

1. ☐ elementary 2. ☐ high school
3. ☐ junior college 4. ☐ college graduate
5. ☐ masters degree 6. ☐ phd

D. From which of the following sources did you obtain your copy of *Ahoy!*?

1. ☐ newsstand 2. ☐ subscription (mail)
3. ☐ from a friend or family member
4. ☐ other ______________________

E. If not currently a subscriber, do you plan to become one?

1. ☐ yes 2. ☐ no

Name____

Street____

City________________ State________________ Zip________________

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 65 MT. MORRIS, ILL.

POSTAGE WILL BE PAID BY ADDRESSEE

Ahoy!

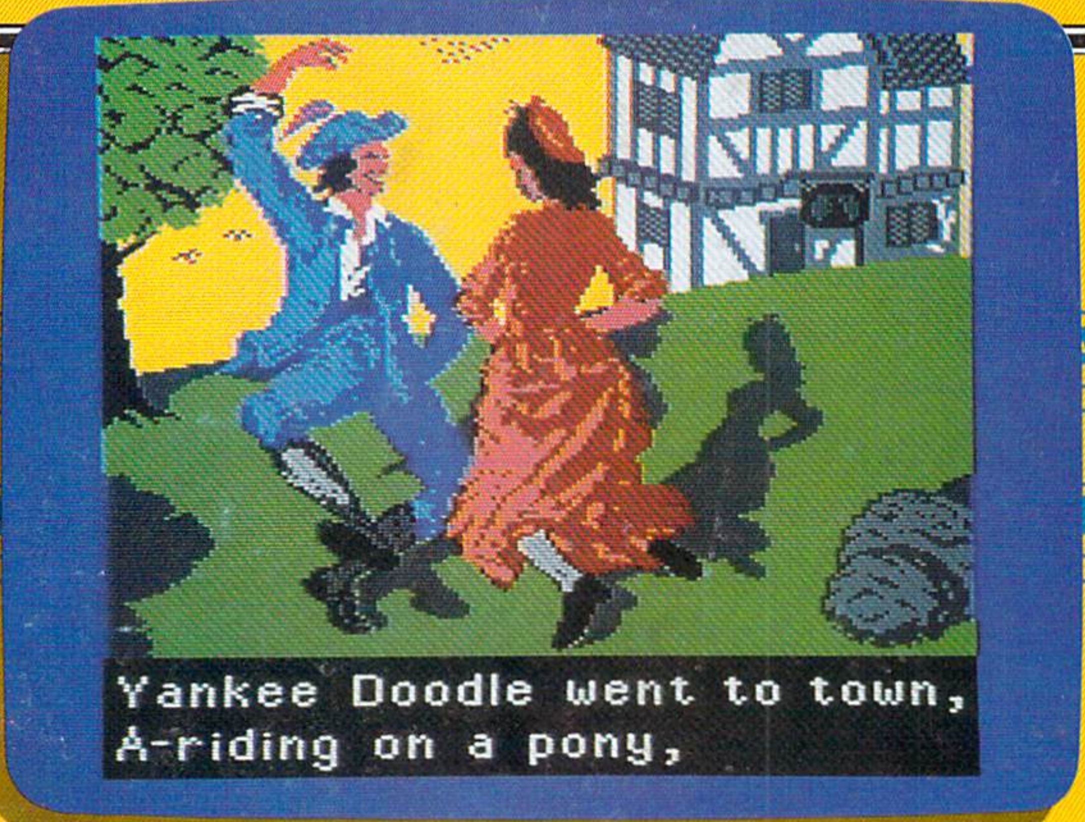
P.O. BOX 341
MT. MORRIS, ILL. 61054

PLACE
STAMP
HERE

AHOY! MAGAZINE
P.O. Box 2501
Clinton, Iowa 52735



THERE'S NO BEATING
MASTERTRONIC
MASTERTRONIC
The GATES of DAWN
NEW
NEW
Zzzz
Featuring instantly drawn animated graphics, joystick driven icons and classic text input. This unusual adventure takes you into the dream world of Sleep. You wander through the 50 weird and wonderful locations seeking an escape from your nightmare.
Gates of Dawn
Now, for the first time, GATES OF DAWN will release you into the world of the dream. You must travel through these timeless passages as the knight of a bygone age. Much of what you see will not be as it appears and conversely much will happen that you cannot always see.
● Let this Arcade Adventure take you into The World of Dreams and Beyond
● 64 Screens of Mystery and Challenge
● Atmospheric Sound
● Marvellously Detailed Graphics and Animation
MASTERTRONIC HAVE DONE IT AGAIN!
Bringing you an added dimension for your Commodore 64/128 with these two outstanding programs on Disc Zzzz and The Gates of Dawn for the incredible price of only $9.99 each.
THE MASTERTRONIC PLEDGE
We promise to bring you two outstanding added dimension programs every month.
Mastertronic International Inc. 7311B Grove Road, Frederick, Maryland 21701 Tel: (301) 695 8877



Yankee Doodle went to town,
A-riding on a pony,